Linear and non-linear high order accurate residual distribution schemes for the discretization of the steady compressible Navier–Stokes equations

Abgrall, Rémi ; de Santis, Dante

Abstract: A robust and high order accurate Residual Distribution (RD) scheme for the discretization of the steady Navier–Stokes equations is presented. The proposed method is very flexible: it is formulated for unstructured grids, regardless the shape of the elements and the number of spatial dimensions. A continuous approximation of the solution is adopted and standard Lagrangian shape functions are used to construct the discrete space, as in Finite Element methods. The traditional technique for designing RD schemes is adopted: evaluate, for any element, a total residual, split it into nodal residuals sent to the degrees of freedom of the element, solve the non-linear system that has been assembled and then iterate up to convergence. The main issue addressed by the paper is that the technique relies in depth on the continuity of the normal flux across the element boundaries: this is no longer true since the gradient of the state solution appears in the flux, hence continuity is lost when using standard finite element approximations. Naive solution methods lead to very poor accuracy. To cope with the fact that the normal component of the gradient of the numerical solution is discontinuous across the faces of the elements, a continuous approximation of the gradient of the numerical solution is recovered at each degree of freedom of the grid and then interpolated with the same shape functions used for the solution, preserving the optimal accuracy of the method. Linear and non-linear schemes are constructed, and their accuracy is tested with the method of the manufactured solutions. The numerical method is also used for the discretization of smooth and shocked laminar flows in two and three spatial dimensions.

DOI: https://doi.org/10.1016/j.jcp.2014.11.031

Linear and non-linear high order accurate residual distribution schemes for the discretization of the steady compressible Navier-Stokes equations

R. Abgrall[a,*], D. De Santis [1b]

[a] *Institut für Mathematikv Universität Zürich Winterthurerstrasse 190 CH-8057 Zürich, Switzerland*
[b] *INRIA Bordeaux–Sud-Ouest, Équipe-projet Bacchus, 200 avenue de la Vieille Tour, 33405 Talence Cedex, France*

Abstract

A robust and high order accurate Residual Distribution (RD) scheme for the discretization of the steady Navier-Stokes equations is presented. The proposed method is very flexible: it is formulated for unstructured grids, regardless the shape of the elements and the number of spatial dimensions. A continuous approximation of the solution is adopted and standard Lagrangian shape functions are used to construct the discrete space, as in Finite Element methods. The traditional technique for designing RD schemes is adopted: evaluate, for any element, a total residual, split it into nodal residuals sent to the degrees of freedom of the element, solve the non linear system that has been assembled and then iterate up to convergence. The main issue addressed by the paper is that the technique relies in depth on the continuity of the normal flux across the element boundaries: this is no longer true since the gradient of the state solution appears in the flux, hence continuity is lost when using standard finite element approximations. Naive solution methods lead to very poor accuracy. To cope with the fact that the normal component of the gradient of the numerical solution is discontinuous across the faces of the elements, a continuous approximation of the gradient of the numerical solution is recovered at each degree of freedom of the grid and then interpolated with the same shape functions used for the solution, preserving the optimal accuracy of the method. Linear and non-linear schemes are constructed, and their accuracy is tested with the method of the manufactured solutions. The numerical method is also used for the discretization of smooth and shocked laminar flows in two and three spatial dimensions.

Keywords: High order schemes, residual distribution, gradient reconstruction, Navier-Stokes equations, compressible flows, unstructured meshes.

1. Introduction

In the last years, different schemes for the numerical solution of compressible flows with higher (more than two) order of accuracy have been proposed. In case of smooth flows, the effectiveness of higher order schemes over standard (theoretically) second order Finite Volume (FV) methods has been showed by several researchers [37]. The attention here is focused on numerical methods for unstructured grids, that are more flexible and have a better chance to be applied to problems relevant for industrial applications. [3]

Probably, one of the most popular scheme in the higher order community is the Discontinuous Galerkin (DG) method [11], which is very attractive for its compactness and flexibility. On the other hand DG methods suffer from the serious drawback of a very fast growth of the number of degrees of freedom (DOFs) with the cell polynomial degree, making these schemes very expensive. Furthermore, the treatment of discontinuities in the DG framework is still not clear and the non-oscillatory properties of DG schemes are still not well understood. An interesting alternative

[1] Current address: William F. Durand Building, 496 Lomita Mall, Stanford, CA 94305-4035, USA, email:ddesanti@stanford.edu
[*] Corresponding author
Email addresses: remi.abgrall@math.uzh.ch (R. Abgrall), ddesanti@stanford.edu (D. De Santis)

[3] Note that block structured meshes can easily be cast into this framework. Instead of the implicit connectivity of the mesh given by the regular indexing of the mesh points, one can explicitely describes the hexahedrons of the meshes.

to DG schemes is represented by Residual Distribution (RD) methods. In the RD framework the solution is continuous and the number of DOFs is much lower than DG methods for the same order of approximation of the solution.

The basic idea of the RD approach is to define for each element an integral quantity (the total residual) which is the integral over the element of the governing equations. Such quantity is then distributed to the DOFs of the elements. The way the distribution process is performed determines the behavior of the numerical scheme. For example, in the past, truly multidimensional schemes have been developed that are defined in such a way that, for each element, only upwind nodes can receive a contribution in the distribution process, while downwind nodes receive no contribution [13]. Due to this multidimensional character, these schemes are much more accurate that standard second-order FV methods. However, such a process has been constructed for scalar problems and the extension to systems of equation lacks physical justification. In addition, the construction of multidimensional upwind schemes is clear only for simplex elements and for the second order of accuracy.

Another complication that arises in the RD frameworks is the discretization of viscous terms. The main issue is related to the necessity to take into account within the same scheme advective terms, by the means of an upwind mechanism, and diffusive phenomena, which on the other hand have an isotropic behavior. Indeed the main idea of RD schemes is to work with the residual of the *whole* PDE: this guaranties formal accuracy, while simpler splitting methods would spoil the accuracy in the framework of RD scheme. This can be seen as a constraint, in practice it is also a guaranty of robustness. Note this is a common practice for continuous Finite Element methods, as well as for several DG methods. To address this problem mixed schemes have been developed, in which RD methods for the advection terms are combined with the Galerkin discretization of the diffusion terms. For such type of schemes, a proper blending between the RD and the Galerkin schemes must be constructed otherwise the accuracy of the resulting schemes is spoiled when advection and diffusion are equally important [25, 28].

Since viscous flows are characterized by the presence, at the same time, of regions dominated by diffusion (*i.e.*, boundary layers), regions where advective phenomena prevail and intermediate zones where advection and diffusion are equally important, it is hazardous to rely on numerical schemes composed by terms that individually might have a different behavior in each region of the domain. Instead of considering two different schemes, one for the advection and one for diffusion terms, here a RD scheme is developed for the whole advection-diffusion equation. This approach introduces a new complication because, for polynomial piecewise approximation of the solution, the normal component of the gradient of the numerical solution is discontinuous on the face of two adjacent elements. A simple fix would be the introduction of a numerical flux for the viscous term, and hence new parameters some times difficult to adjust. Instead of actually considering a numerical flux along the faces of the elements, as in DG or FV schemes, the approach adopted in this work consists in recovering a unique set of values for the gradient of the numerical solutions at the DOFs of the grid. Then, these values are interpolated with the same continuous functions used to interpolate the solution. It is evident that the gradients have to be recovered with higher order of accuracy to construct a high order scheme: the crucial point is the strategy used to recover the gradients at the DOFs.

A RD method based on a similar idea was proposed in [10] for triangular and tetrahedral elements at the third order accuracy. The main idea in [10], was the use of the Green-Gauss gradient reconstruction at the grid vertices employed in the discretization of the viscous terms and also used to obtain a higher order approximation of the solution on each edge of the elements, through the use of Hermite interpolation. One of the main drawback of this approach is that the accuracy of the numerical solution is deeply related to the quality of the mesh. Furthermore, it is not clear how to extend (if possible) this approach to orders more than three and to generic elements.

In this work, within the RD framework, we describe an alternative discretization to the classical multidimensional upwind schemes which could allow an easy formulation also on not-simplex elements and to generic order of accuracy. We also describe how to construct linear and non-linear schemes. While linear schemes guarantee a better accuracy, non-linear schemes are necessary when discontinuities are present in the solution. Both schemes rely on a high order accurate reconstruction of the gradient of the numerical solution, which is the key element to preserve the global accuracy of the numerical scheme in the whole advection-diffusion regime. To maintain the compactness and the efficiency of the numerical scheme, the least square or the L^2-projection methods for the gradient reconstruction are discarded in favor of a compact and very accurate technique, the so-called Super-convergent Patch Recovery method of Zienkiewicz and Zhu [38, 39]. This has already been used by the authors in the context of the RD methods, for the discretization of scalar advection-diffusion problems [3]

The structure of the paper is as follows. In Section 2, the multidimensional Navier-Stokes equations for laminar flows are introduced. In Section 3, the proposed RD method for advection-diffusion problems is described. The detail

of the construction of linear and non-linear schemes are given, together with the description of the imposition of the boundary conditions. The theoretical description of the numerical method is kept general, regardless the shape of the elements and the order of accuracy used. In Section 4, the strategy used to reconstruct the gradient is described in detail and comparisons in term of accuracy, between different gradient reconstruction techniques, are given. In Section 5, is described the construction of a matrix-free method for a robust and effective implicit scheme. In Section 6, numerical experiments are reported, here we restrict ourselves to second and third order accurate numerical simulations on triangle or tetrahedral elements. The method of the manufactured solutions is used to assess the accuracy of the numerical schemes in a wide range of Reynold numbers. The simulation of smooth flows over a two dimensional NACA-0012 airfoil, a flat plate at high Reynolds and a three dimensional delta wing are performed with the linear scheme, while a shock-boundary layer interaction problem is used to test the non-oscillatory properties of the non-linear scheme. In Section 7, final remarks are given.

2. Governing equations

The equations of motion considered in this work are the compressible Navier-Stokes equations. For unsteady flows, they write:

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla}\cdot \boldsymbol{m} = 0 \\
&\frac{\partial \boldsymbol{m}}{\partial t} + \boldsymbol{\nabla}\cdot \left(\frac{\boldsymbol{m}\otimes\boldsymbol{m}}{\rho} + P\mathbb{I} \right) = \boldsymbol{\nabla}\cdot \mathbb{S} \\
&\frac{\partial E^t}{\partial t} + \boldsymbol{\nabla}\cdot \left((E^t + P)\frac{\boldsymbol{m}}{\rho} \right) = \boldsymbol{\nabla}\cdot \left(\mathbb{S}\cdot\frac{\boldsymbol{m}}{\rho} - \boldsymbol{q} \right)
\end{aligned} \tag{1}$$

where the unknowns are the density ρ, the momentum vector $\boldsymbol{m}$ and the the total energy per unit volume E^t, with

$$\boldsymbol{m} = \rho \boldsymbol{v} \quad \text{and} \quad E^t = \rho e + \frac{\|\boldsymbol{m}\|^2}{2\rho}.$$

Here, $\boldsymbol{v}$ is the velocity vector and e is the specific (*i.e.*, for unit of mass) internal energy. Furthermore, in the system (1) P is the static pressure, $\mathbb{I} \in \mathbb{R}^{N_{\text{dim}}}$ is the identity matrix, with N_{dim} number of spatial dimensions, $\mathbb{S}$ is the viscous stress tensor and $\boldsymbol{q}$ is the heat flux vector. For a Newtonian flow, under the Stokes hypothesis, the stress tensor can be written as

$$\mathbb{S} = -\frac{2}{3}\mu(\boldsymbol{\nabla}\cdot\boldsymbol{v})\mathbb{I} + \mu(\boldsymbol{\nabla}^{\mathrm{T}}\boldsymbol{v} + \boldsymbol{\nabla}\boldsymbol{v}).$$

where μ is the dynamic viscosity coefficient.

The heat flux, using the Fourier law, can be written as: $\boldsymbol{q} = -\kappa\boldsymbol{\nabla}T$, where the thermal conductivity κ is evaluated using the constant Prandtl number hypothesis

$$\kappa = c_p \frac{\mu}{Pr},$$

with c_p the specific heat at constant pressure and Pr the Prandtl number.

The system (1) is closed by providing a thermodynamic model for the fluid. In this work only the polytropic ideal gas model is considered, for which the pressure and the temperature can be written as

$$P(e,\rho) = (\gamma - 1)\rho e, \quad \text{and} \quad T(e) = \frac{(\gamma-1)}{\mathcal{R}}e,$$

with $\gamma = c_p/c_v$ the specific heats ratio and $\mathcal{R}$ the gas constant.

If the vector of the conservative variables $\mathsf{u} \in \mathbb{R}^{N_{\text{eq}}}$ is introduced, with N_{eq} the number of equations, the system of equations (1) can be written in a more compact form as follows

$$\frac{\partial \mathsf{u}}{\partial t} + \boldsymbol{\nabla}\cdot\mathbf{f}^a(\mathsf{u}) = \boldsymbol{\nabla}\cdot\mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}), \tag{2}$$

with

$$\mathsf{u} = \begin{pmatrix} \rho \\ \boldsymbol{m} \\ E^t \end{pmatrix}, \qquad \mathbf{f}^a(\mathsf{u}) = \begin{pmatrix} \boldsymbol{m} \\ \dfrac{\boldsymbol{m}\otimes\boldsymbol{m}}{\rho} + P\mathbb{I} \\ (E^t + P)\dfrac{\boldsymbol{m}}{\rho} \end{pmatrix}, \qquad \mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \begin{pmatrix} 0 \\ \mathbb{S} \\ \mathbb{S}\cdot\dfrac{\boldsymbol{m}}{\rho} + \kappa\boldsymbol{\nabla}T \end{pmatrix},$$

where $\mathbf{f}^a$ and $\mathbf{f}^v$ are the advective and the viscous flux functions, respectively. For later convenience, it is pointed out here that, since the viscous flux function is homogeneous of degree one with respect to the gradient of the conservative variables, it can be also written as follows

$$\mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}) = \mathbb{K}(\mathsf{u})\boldsymbol{\nabla}\mathsf{u},$$

where the viscous flux Jacobian $\mathbb{K}(\mathsf{u}) \in \mathbb{R}^{N_{\text{eq}}\times N_{\text{dim}}\times N_{\text{eq}}\times N_{\text{dim}}}$ is defined as

$$\mathbb{K}(\mathsf{u}) = \frac{\partial \mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})}{\partial \boldsymbol{\nabla}\mathsf{u}}, \qquad \text{with} \qquad \mathsf{K}_{ij}(\mathsf{u}) = \frac{\partial \mathsf{f}_i^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u})}{\partial\left(\dfrac{\partial \mathsf{u}}{\partial x_j}\right)}.$$

This paper consideres only the steady case, i.e. when there is no time dependancy in the Navier-Stokes equations: we are interested in

$$\boldsymbol{\nabla}\cdot\mathbf{f}^a(\mathsf{u}) = \boldsymbol{\nabla}\cdot\mathbf{f}^v(\mathsf{u}, \boldsymbol{\nabla}\mathsf{u}), \tag{3}$$

with Dirichlet boundary conditions on the velocity and Neuman conditions on the temperature on solid walls, and inflow/out flow conditions at infinity. Dirichlet conditions on the wall temperature could also be dealt with but have not been considered in this paper. The equations are made dimensionless by taking as reference quantities the density, the speed of sound and the temperature at the free-stream and a characterizing length of the problem.

3. RD space discretization

In this section the RD space discretization of the steady Navier-Stokes equations is described. The numerical method used to discretize the system of equations follows from the scheme already proposed by the authors in [3] to discretize the scalar advection-diffusion problem.

The computational domain Ω is discretized with N_e non-overlapping elements with characteristic length h, the set of all the elements is denoted by $\mathcal{E}_h$, the list of the DOFs is denoted by Σ_h, and the total number of DOFs is N_{dof}. The solution is approximated on each element by k-th order polynomials which are assumed to be continuous within the elements and on the faces of the elements. In this work only Lagrangian basis function are used, see [8] for an example of RD scheme using non-Lagrange elements. The approximated solution u^h can be written as

$$\mathsf{u}^h(\boldsymbol{x}) = \sum_{i\in\Sigma_h} \psi_i(\boldsymbol{x})\,\mathsf{u}_i, \quad \boldsymbol{x}\in\Omega,$$

with $\mathsf{u}_i = \mathsf{u}(\boldsymbol{x}_i)$ and ψ_i the i-th Lagrangian basis function. The order k of approximation of the solution is left unspecified, since the theoretical construction of the numerical scheme is shown for any order of approximation. It is worth also noticing that the theoretical description of the numerical scheme is non limited just to simplex elements.

The approximated solution, in general, will not satisfy the governing equations, this means that the integral of the Eq. (3) calculated on each element e will be not null, but it will give rise to a residual on each element, namely

$$\Phi^e(\mathsf{u}^h, \boldsymbol{\nabla}\mathsf{u}^h) = \int_{\Omega_e} \boldsymbol{\nabla}\cdot\left(\mathbf{f}^a(\mathsf{u}^h) - \mathbf{f}^v(\mathsf{u}^h, \boldsymbol{\nabla}\mathsf{u}^h)\right)\mathrm{d}\Omega, \tag{4}$$

where

$$\boldsymbol{\nabla}\mathsf{u}^h|_e = \sum_{i\in\Sigma_h^e} \boldsymbol{\nabla}\psi_i\,\mathsf{u}_i,$$

is the internal gradient of the element e, computed with the gradient of the shape functions as in the Finite Element approach, and Σ_h^e is the set of DOFs of the element e. The integral quantity $\Phi^e(\mathsf{u}^h, \boldsymbol{\nabla}\mathsf{u}^h)$ is called total residual of the

element e. To keep the description of the scheme simple, the contribution to the residual coming from the boundary conditions is not considered here, but it will be described later.

In practice, the total residual is never computed as reported in Eq. (4), the volume integral is transformed instead into a surface integral by the means of the divergence theorem. This procedure is legitimate for the divergence of the advective flux, since the numerical solution by definition is continuous across the face of two neighbor elements, however, in handling the divergence of the viscous flux one has to cope with the fact that the normal component of internal gradient of the numerical solution is discontinuous across the face of two adjacent elements. The continuity of the normal flux across the element boundary is one of the key arguments used in [6] to show the consistency of the numerical scheme with respect to the original PDE. One way to overcome this difficulty would be to introduce a numerical a flux for the viscous contribution, in the spirit of [2], leading to the introduction of additional parameters that might be difficult to understand, as well as, very likely, tricky accuracy issues of the same kind as the one we are challenging in this paper. A numerical flux for viscous problems in the second order formulation was already proposed in [15] in the context of FV and DG methods, and could be also applicable in principle to RD schemes. This approach however deserves further studies which are beyond the scope of this work.

Suppose that $[\widetilde{\boldsymbol{\nabla}\mathsf{u}}_i]_{i=1,\dots,N_{\text{dof}}}$ is a set of unique values of the gradient of the numerical solution. Hence the gradient can be interpolated with the same shape functions used for the solution, namely

$$\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}(\boldsymbol{x}) = \sum_{i\in\Sigma_h} \psi_i(\boldsymbol{x})\, \widetilde{\boldsymbol{\nabla}\mathsf{u}}_i, \quad \boldsymbol{x}\in\Omega.$$

The quantity $\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}$ is now continuous on the faces of the elements and the total residual can be reformulated as follows

$$\begin{aligned}\Phi^e &= \oint_{\partial\Omega_e} \left(\mathbf{f}^a(\mathsf{u}^h) - \mathbf{f}^v(\mathsf{u}^h, \widetilde{\boldsymbol{\nabla}\mathsf{u}^h})\right)\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega \\ &= \oint_{\partial\Omega_e} \left(\mathbf{f}^a(\mathsf{u}^h) - \mathbb{K}(\mathsf{u}^h)\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right)\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega.\end{aligned} \tag{5}$$

The procedure to recovery the gradient at each DOF, *i.e.*, how compute $[\widetilde{\boldsymbol{\nabla}\mathsf{u}}_i]_{i=1,\dots,N_{\text{dof}}}$, is detailed in Section 4.

Note that the idea to use a recovery procedure to remove the discontinuities at the element interfaces is not new and has been also used in the context of DG methods for solving viscous flow problems, see e.g. [21, 36]. In these works, starting from the piece-wise discontinuous solution, a continuous approximation of the solution at the cell interface is reconstructed on the union of adjacent cells to the interface, and such continuous solution is used to evaluate the viscous flux function at the element interface. In [14] the reconstruction procedure has been used in the context of the DG methods to construct a class of schemes, called $\mathrm{P}_N\mathrm{P}_M$, to solve the Navier-Stokes equation using the diffusive generalized Riemann problem [15] to compute the numerical viscous fluxes at the interface.

In order to handle only nodal values, the total residual (5) is first distributed, in some way that we describe later in the text, to the DOFs of the element as follows

$$\Phi^e_i = \beta^e_i(\mathsf{u}^h)\,\Phi^e(\mathsf{u}^h, \widetilde{\boldsymbol{\nabla}\mathsf{u}^h}), \quad \forall i\in\Sigma^e_h, \tag{6}$$

where β^e_i are the distribution coefficients, which can be in general function of u^h. It is easy to see that the nodal residuals, Φ^e_i, must satisfy the following conservation constraint [6]

$$\sum_{i\in\Sigma^e_h} \Phi^e_i = \Phi^e, \quad \forall e\in\mathcal{E}_h. \tag{7}$$

Following [6], formal accuracy is achieved when, for a smooth exact solution u_{exact} which is interpolated by u^h, the residuals Φ^e_i evaluated for that particular function satisfy

$$\Phi^e_i(\mathsf{u}^h) = O(h^{k+N_{\text{dim}}}), \tag{8}$$

where N_{dim} is the geometrical dimension ($N_{\text{dim}} = 2, 3$) and k is the polynomial degree used to interpolate the solution. Note that the estimate (8) uses in depth the structure of the problem: we are solving a *steady* problem. The form

(6) satisfies this constraint provided that the β_i^e are uniformly bounded with respect to the numerical solution and the mesh. Following Roe, if a set of residuals satisfies (6), then the scheme is said to be linearity preserving, even if we are seeking for an accuracy larger that second order.

To obtain an equation for each nodal value of the numerical solution, the following relations are written for each DOF

$$\sum_{e \in \mathcal{E}_h^i} \Phi_i^e = 0, \quad \forall i \in \Sigma_h, \tag{9}$$

where $\mathcal{E}_h^i$ is the set of the elements which share the DOF i. The previous relations define a set of non-linear equations that must be solved for the nodal values of the solution $[\mathsf{u}_i]_{i=1,\dots,N_{\text{dof}}}$. In practice, the solution with a RD method is obtained by the means of an iterative method, which in the simplest form reads

$$\frac{\mathsf{u}_i^{n+1} - \mathsf{u}_i^n}{\Delta t_i^n} = -\sum_{e \in \mathcal{E}_h^i} \Phi_i^e, \quad \forall i \in \Sigma_h, \tag{10}$$

with Δt_i^n a scaled pseudo-time step. The change of the nodal values of the solution during the iterative process is driven by the non-zero total residuals on the elements; for $n \to \infty$ the total residual on each element vanishes and the steady state solution is obtained.

Remark 3.1 (About the accuracy to which approximate $\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}$)**.** *A close examination of* (5) *combined with* (7) *and compared with* (8) *helps to understand the kind of difficulties we are trying to solve. From* (5)*, we ask that*

$$\oint_{\partial\Omega_e} \left(\mathbf{f}^a(\mathsf{u}^h) - \mathbb{K}(\mathsf{u}^h)\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega = O(h^{k+N_{\text{dim}}}).$$

Since the problem is steady, we have indeed, since $\oint_{\partial\Omega_e} \left(\mathbf{f}^a(\mathsf{u}) - \mathbb{K}(\mathsf{u})\boldsymbol{\nabla}\mathsf{u}\right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega = 0$*,*

$$\begin{aligned}
&\oint_{\partial\Omega_e} \left(\mathbf{f}^a(\mathsf{u}^h) - \mathbb{K}(\mathsf{u}^h)\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega = \\
&\quad \oint_{\partial\Omega_e} \left(\left[\mathbf{f}^a(\mathsf{u}^h) - \mathbf{f}^a(\mathsf{u})\right] - \left[\mathbb{K}(\mathsf{u}^h)\widetilde{\boldsymbol{\nabla}\mathsf{u}^h} - \mathbb{K}(\mathsf{u})\boldsymbol{\nabla}\mathsf{u}\right]\right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega \\
&\qquad = O(h^{k+N_{\text{dim}}}) + \oint_{\partial\Omega_e} \left(\left[\mathbb{K}(\mathsf{u}^h)\widetilde{\boldsymbol{\nabla}\mathsf{u}^h} - \mathbb{K}(\mathsf{u})\boldsymbol{\nabla}\mathsf{u}\right]\right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega
\end{aligned}$$

because for regular solutions, $\mathbf{f}^a(\mathsf{u}^h) - \mathbf{f}^a(\mathsf{u}) = O(h^{k+1})$ *and* $|\partial\Omega_e| = O(h^{N_{\text{dim}}-1})$*. Using* (8)*, this implies that we must have*

$$\oint_{\partial\Omega_e} \left(\left[\mathbb{K}(\mathsf{u}^h)\widetilde{\boldsymbol{\nabla}\mathsf{u}^h} - \mathbb{K}(\mathsf{u})\boldsymbol{\nabla}\mathsf{u}\right]\right) \cdot \boldsymbol{n} \, \mathrm{d}\partial\Omega = O(h^{k+N_{\text{dim}}})$$

and the easiest way to get this estimate is that

$$\widetilde{\boldsymbol{\nabla}\mathsf{u}} - \boldsymbol{\nabla}\mathsf{u}^h = O(h^{k+1}).$$

This is not immediate to get because since $\mathsf{u}^h - \mathsf{u} = O(h^{k+1})$ *from the approximation theory, standard results only say that* $\widetilde{\boldsymbol{\nabla}\mathsf{u}} - \widetilde{\boldsymbol{\nabla}\mathsf{u}^h} = O(h^k)$*.*

3.1. Construction of distribution schemes for advection-diffusion problems

The distribution process in Eq. (6) has been expressed through the use of generic distribution coefficients, the focus now is on how to actually construct a distribution scheme for the Navier-Stokes equations. In this work, the attention is focused on the construction of centered schemes. Linear and non-linear schemes are considered.

It is worth clarifying what we call here *centered RD schemes*. RD multidimensional upwind schemes, *for scalar equations*, are construct such that on each element only upwind nodes (upwind with respect to an average advection speed on the element) get a non-zero nodal residuals in the distribution process.

As opposed to multidimensional upwind schemes, one can construct a numerical method in which all DOFs of the element receive a contribution in the distribution process. The most simple scheme of this type that one could construct is a centered scheme based on a purely isotopic distribution of the total residual on each element. However, this would result in an unstable numerical scheme. It is important that the numerical scheme includes also some biased terms beside the centered ones. For this reason, in this work the term "centered" is used to indicate a biased scheme made of an isotropic part plus some streamline term. It is not used to convey the idea of a numerical scheme in which upwind or bias are not included at all. In this sense, finite element approaches like SUPG [17] or Lax-Wendroff [32] can be interpreted as centered RD schemes.

We want to remark here, that the theoretical description of the numerical scheme is kept as general as possible, and we do not specialize it to a particular order of discretization or type of element.

3.1.1. Linear scheme

The linear scheme proposed in this work is the extension to the integral formulation of the classical Ni's Lax-Wendroff scheme [23], this scheme has been also successfully used in the context of the RD schemes to discretize the advection-diffusion scalar equation, see [3]. The nodal residual, for the DOF i of the element e, can be written as follows

$$\Phi_i^e = \frac{\Phi^e}{N_{\text{dof}}^e} + \int_{\Omega_e} \mathbf{A}\cdot\boldsymbol{\nabla}\psi_i \,\Xi\left(\mathbf{A}\cdot\boldsymbol{\nabla}\mathsf{u}^h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right)\right)\mathrm{d}\Omega, \tag{11}$$

where N_{dof}^e is the number of DOFs of the element e, $\mathbf{A} = \partial\mathbf{f}^a/\partial\,\mathsf{u}$ is the inviscid flux Jacobian, and Ξ is a scaling matrix which is taken as

$$\Xi = \frac{1}{N_{\text{dim}}}|\Omega_e|\left(\sum_{i=1}^{N_{\text{dof}}^e} \mathsf{R}_{n_i}(\bar{\mathsf{u}})\,\Lambda_{n_i}^+(\bar{\mathsf{u}})\,\mathsf{L}_{n_i}(\bar{\mathsf{u}}) + \sum_{j=1}^{N_{\text{dim}}} \mathsf{K}_{jj}(\bar{\mathsf{u}})\right)^{-1}. \tag{12}$$

In (12), $\bar{\mathsf{u}}$ is the arithmetic average of the vector of the conservative variables on the element, R_n, L_n are respectively the matrices of the right and left eigenvectors along the direction of the generic vector $\boldsymbol{n}$, and $\Lambda_n = \text{diag}(\lambda_n)$ is the corresponding diagonal matrix of the eigenvalues. The operator $(\cdot)^+$ selects only the positive values and sets the negative ones to zero. The vector $\boldsymbol{n}_i$ is taken as follows

$$\boldsymbol{n}_i = \frac{1}{N_{\text{dim}}}\int_{\Omega_e} \boldsymbol{\nabla}\psi_i \,\mathrm{d}\Omega,$$

such that it has the dimensions of a length (surface) in two (three) spatial dimensions. This definition is quite general, and has been chosen in order to have an unique definition of the vector, regardless the type of the element and order of approximation used.

The scheme is consistent since it achieves the conservation constraint (7). Formal accuracy is satisfied too: when the exact solution is injected in Eq. (12), the residual is zero [4] because the total residual vanishes, and the integral term vanishes as well due to the fact that the right hand side term in integral is exactly the governing equation. Note that although the integral in Eq. (11) is strictly computed on the element interior, for sake of robustness and consistency with the previous derivation of the RD space discretization, the reconstructed gradients are used in place of the internal gradients in the calculation of viscous terms.

The scheme is linearity preserving but not positive, which means that it is not guaranteed that it is monotone for discontinuous solutions. Due to the integral formulation, it is valid for any type of element and for any order of approximation of the solution.

3.1.2. Non-linear scheme

Non-linear schemes are needed to combine the non-oscillatory behavior with the higher order discretization. The basic idea to construct a non-linear scheme is to start with a first order monotone method, and to map its nodal residuals onto a set of positive and non-linear residuals [1, 3]. The first step to construct a non-linear RD method is

[4] Up to truncation errors because it is the Lagrange interpolant of the exact solution that plays a role in the discrete formulation.

to start with a first-order and monotone scheme. Any monotone first-order scheme could be used, but in this work we consider only the Rusanov' scheme (also known as the local Lax-Friedrichs scheme) defined as

$$\tilde{\Phi}_i^e = \frac{\Phi^e}{N_{\text{dof}}^e} + \frac{1}{N_{\text{dof}}^e}\,\alpha^e \sum_{\substack{j\in\Sigma_h^e \\ j\neq i}} (\mathsf{u}_i - \mathsf{u}_j), \qquad \forall i \in \Sigma_h^e,$$

with α^e large enough for stability, in practice it is taken as

$$\alpha^e = \max_{j\in\Sigma_h^e} \left(|\lambda_{n_j}| + \left(\frac{\mu}{\rho} + \frac{\kappa}{c_v} \right)_j \right).$$

Since the Rusanov' scheme is first order accurate, its distribution coefficients, β_i^e, are unbounded. The construction of the non-linear scheme consists in mapping the distribution coefficients of the low order scheme onto non-linear bounded distribution coefficients. As described in [1], the mapping for a system of equations is constructed by projecting the residual of the first-order scheme onto the space of the left eigenvectors, such that

$$\Phi_i^{e\star} = \mathsf{L}_n \tilde{\Phi}_i^e \qquad \text{and} \qquad \Phi^{e\star} = \sum_{i\in\Sigma_h^e} \Phi_i^{e\star} = \mathsf{L}_n \Phi^e,$$

where the mean fluid velocity vector on the element is used as direction vector for the computation of the eigenvectors. The nodal high order residuals are obtained as follows

$$\hat{\Phi}_i^{e\star} = \hat{\beta}_i^{e\star} \Phi^{e\star},$$

with the bounded distribution coefficients, $\hat{\beta}_i^{e\star}$, obtained by applying a non-linear mapping to the original unbounded distribution coefficients, $\beta_i^{e\star} = \Phi_i^{e\star}/\Phi^{e\star}$. The map is constructed as follows

$$\hat{\beta}_i^{e\star} = \frac{\left(\dfrac{\Phi_i^{e\star}}{\Phi^{e\star}} \right)^+}{\displaystyle\sum_{j\in\Sigma_h^e} \left(\frac{\Phi_i^{e\star}}{\Phi^{e\star}} \right)^+},$$

which correspond to the so-called PSI limiter. Finally, the high order nodal residuals are projected back on the physical space: $\hat{\Phi}_i^e = \mathsf{R}_n \hat{\Phi}_i^{e\star}$.

The use of a scheme, like the Rusanov's scheme, in combination with the limiting technique, produces undamped spurious modes and a poor iterative convergence to the steady state solution [1]. This is due to the fact that the method could be locally downwind. The cure to this problem consists in adding a filtering term by means of a streamline dissipation term

$$\Phi_i^e = \hat{\Phi}_i^e + \varepsilon_h^e(\mathsf{u}^h) \int_{\Omega_e} \Big(\mathbf{A}\cdot\boldsymbol{\nabla}\psi_i - \boldsymbol{\nabla}\cdot(\mathbb{K}\boldsymbol{\nabla}\psi_i) \Big)\, \Xi \Big(\mathbf{A}\cdot\boldsymbol{\nabla}\mathsf{u}^h - \boldsymbol{\nabla}\cdot(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}) \Big)\, \mathrm{d}\Omega. \tag{13}$$

where the scaling matrix Ξ is defined as in Eq. (12). The term $\varepsilon_h^e(\mathsf{u}^h)$ is a smoothness sensor which assures that the filtering term is added only in the smooth regions of the solution, namely $\varepsilon_h^e(\mathsf{u}^h) \sim 1$ in smooth regions and $\varepsilon_h^e(\mathsf{u}^h) \sim 0$ near discontinuities. A pressure-based shock sensor is used for $\varepsilon_h^e(\mathsf{u}_h)$, see [4],

$$\varepsilon_h^e(\mathsf{u}_h) = 1 - \max_{i\in\Sigma_h^e} \left(\max_{e\in\mathcal{E}_h^i} \left(\frac{|P_i - \bar{P}_e|}{|P_i| + |\bar{P}_e| + \epsilon} \right) \right),$$

where P_i is the pressure at the DOF i, $\bar{P}_e$ is the arithmetic average of the pressure on the element e, and $\epsilon = 10^{-12}$.

Remark 3.2. *In FE methods, one uses the weak form of the equations and the integration by part to reduce the order the of derivatives. In the RD approach we work with the strong form of the governing equation and using the continuity of the solution and of the gradient (after the reconstruction) we can put the domain integral of the governing equation*

as integrals over the faces of the elements. In our opinion, this leads to several advantages since there is no need to accurately and separately evaluate surface/volume and *boundary integrals. Indeed, only the boundary integrals need to be accurately evaluated, and the surface ones that might come up need less demanding quadrature formula. We refer to remark 3.3 for more details on this point. It easy to see that the RD approach is significantly less expensive than the FE approach.*

When shocked problems are considered. The FE approach uses artificial viscosity techniques to preserve the monotonicity of the solution, these introduce tuning parameters in the definition of the artificial viscosity and an additional domain integral to compute the contribution of the artificial viscosity to the residual of the scheme. The non-linear approach adopted in the RD method, is, in our opinion, more elegant and effective than the artificial viscosity approach. In fact, in the non-linear approach there are no tuning parameters and monotonicity is automatically enforced in the numerical scheme. The calculation of the non-linear residual is very cheap, it involves only algebraic manipulations with a computational cost similar to the limiting procedure adopted in the FV framework.

3.2. Improved discretization of the diffusion terms

For a simple advection-diffusion problem, the scalar counterparts of the schemes (11) and (13) showed a lack of robustness in the iterative convergence. In order to introduce an improvement in the previous numerical schemes, suppose to write the governing system of equations in the following form

$$\begin{cases} \boldsymbol{\nabla}\cdot\mathbf{f}^a(\mathsf{u}) - \boldsymbol{\nabla}\cdot(\mathbb{K}\mathsf{w}) = 0 \\ \mathsf{w} - \boldsymbol{\nabla}\mathsf{u} = 0 \end{cases} \tag{14}$$

In the previous system, an additional variable w has been introduced for the gradients, in this way the original second order governing equations can be recasted as a first order system. The first order formulation for viscous problems has become a very standard approach in the DG framework, where numerical fluxes for the solution, u, and the gradient variable, w, are introduced to handle the discontinuities at interface of the elements [12]. Here, the discontinuity of the variable w at the faces of the elements is taken into account via the gradient reconstruction procedure.

Consider a numerical scheme for the system (14) of the following type

$$\int_{\Omega_e} \psi_i \begin{pmatrix} \boldsymbol{\nabla}\cdot\mathbf{f}^a(\mathsf{u}^h) - \boldsymbol{\nabla}\cdot(\mathbb{K}\mathsf{w}) \\ \mathsf{w} - \boldsymbol{\nabla}\mathsf{u}^h \end{pmatrix} \mathrm{d}\Omega + \int_{\Omega_e} \mathbb{A}\cdot\boldsymbol{\nabla}\psi_i \, \mathbb{T} \begin{pmatrix} \boldsymbol{\nabla}\cdot\mathbf{f}^a(\mathsf{u}^h) - \boldsymbol{\nabla}\cdot(\mathbb{K}\mathsf{w}) \\ \mathsf{w} - \boldsymbol{\nabla}\mathsf{u}^h \end{pmatrix} \mathrm{d}\Omega = 0, \tag{15}$$

where

$$\mathbb{A}\cdot\boldsymbol{\nabla}\psi_i = \begin{pmatrix} \mathbf{A}\cdot\boldsymbol{\nabla}\psi_i & -\mathbb{K}\cdot\boldsymbol{\nabla}\psi_i \\ -\boldsymbol{\nabla}\psi_i & 0 \end{pmatrix},$$

and the scaling term is taken as follows

$$\mathbb{T} = \begin{pmatrix} \Xi & 0 \\ 0 & \Xi_d \end{pmatrix}.$$

Supposing, now, that the gradient of the numerical solution has been reconstructed at each DOF, one can replace the equations on the second line of the system (15) with the approximation $\mathsf{w} \simeq \widetilde{\boldsymbol{\nabla}\mathsf{u}^h}$ and consider only the first line, which now reads

$$\begin{aligned} \int_{\Omega_e} \psi_i \left(\boldsymbol{\nabla}\cdot\mathbf{f}^a(\mathsf{u}^h) - \boldsymbol{\nabla}\cdot(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}^h})\right)\mathrm{d}\Omega \; + \int_{\Omega_e} \mathbf{A}\cdot\boldsymbol{\nabla}\psi_i \, \Xi\left(\mathbf{A}\cdot\boldsymbol{\nabla}\mathsf{u}^h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right)\right)\mathrm{d}\Omega \\ + \int_{\Omega_e} \mathbb{K}\,\boldsymbol{\nabla}\psi_i \cdot \left(\Xi_d\left(\boldsymbol{\nabla}\mathsf{u}^h - \widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right)\right)\mathrm{d}\Omega = 0. \end{aligned} \tag{16}$$

The first two integrals in the previous equation represent the original discretization for the advection-diffusion problem. The last integral represents an additional stabilization term, for the diffusive terms only, which vanishes in the advective limit, and the parameter matrix Ξ_d is dimensionless. It is interesting to note that the role of this additional term is to penalize the difference between the discontinuous and the reconstructed gradients, on each element. With a slightly different procedure, a similar stabilization term for the diffusive part has been obtained by Nishikawa [24] for the RD discretization of scalar diffusion problems.

While in the scalar case the value of Ξ_d can be set properly [24], in the case of Navier-Stokes equations, a optimal tuning is not possible and for this reason Ξ_d is taken as the identity matrix. Such a choice is justified by the fact that, in numerical experiments, it was observed that the values of Ξ_d affect only marginally the iterative convergence to the steady state of the scheme and they have a very limited influence on the accuracy of the numerical scheme.

With the Eq. (16) in mind, the schemes (11) and (13) are modified in order to include the extra stabilization term for the diffusive part of the equation. Namely, the linear scheme reads

$$\begin{aligned}\Phi_i^e = \frac{\Phi^e}{N_{\text{dof}}^e} &+ \int_{\Omega_e} \mathbf{A}\cdot\boldsymbol{\nabla}\psi_i\, \Xi\left(\mathbf{A}\cdot\boldsymbol{\nabla}\mathsf{u}^h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\boldsymbol{\nabla}\mathsf{u}^h\right)\right)\mathrm{d}\Omega \\ &+ \int_{\Omega_e} \mathbb{K}\,\boldsymbol{\nabla}\psi_i\cdot\left(\boldsymbol{\nabla}\mathsf{u}^h - \widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right)\mathrm{d}\Omega,\end{aligned} \tag{17}$$

while the non-linear scheme becomes

$$\begin{aligned}\Phi_i^e = \hat{\Phi}_i^e &+ \varepsilon_h^e(\mathsf{u}^h)\int_{\Omega_e}\left(\mathbf{A}\cdot\boldsymbol{\nabla}\psi_i - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\boldsymbol{\nabla}\psi_i\right)\right)\Xi\left(\mathbf{A}\cdot\boldsymbol{\nabla}\mathsf{u}^h - \boldsymbol{\nabla}\cdot\left(\mathbb{K}\widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right)\right)\mathrm{d}\Omega \\ &+ \int_{\Omega_e}\mathbb{K}\,\boldsymbol{\nabla}\psi_i\cdot\left(\boldsymbol{\nabla}\mathsf{u}^h - \widetilde{\boldsymbol{\nabla}\mathsf{u}^h}\right)\mathrm{d}\Omega.\end{aligned} \tag{18}$$

Remark 3.3. *The formulation Eq. (18) involves surface/volume integration. It is not necessarily mandatory to use exact quadrature formula since we only need consistency at the level of the elements, thanks to the conservation relation* (7) *and the Lax-Wendroff like theorem shown in [6] which was proved for the pure hyperbolic case, but the extension to the parabolic case is straightforward. What is really needed, as shown in [5], is that the sum of the residual for one element sums up to a total residual that itself can be a discrete approximation of the normal flux integral (including viscous terms). For accuracy reasons only, this quadrature formula has to be of maximum order. Next, the stream-line term, or its discrete version needs to be dissipative: this induce mild constraints on the quadrature formula that can even be non consistent. This set of remarks has been used in the implementation to lower a lot the number of operations. See [5] for more details.*

3.3. Boundary conditions

The imposition of the boundary conditions in the RD framework has not reached yet a mature level of understanding, for this reason it is shown here the way followed in this work to impose the boundary conditions.

Consider the weak form of a generic steady conservation law

$$\int_\Omega \psi\boldsymbol{\nabla}\cdot\mathbf{f}(\mathsf{u}^h)\,\mathrm{d}\Omega = 0,$$

an integration by parts gives

$$\int_{\partial\Omega}\mathbf{f}(\mathsf{u}^\partial)\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega - \int_\Omega \boldsymbol{\nabla}\psi\cdot\mathbf{f}(\mathsf{u}^h)\,\mathrm{d}\Omega = 0,$$

where u^∂ represents the solution state that takes into account the boundary conditions. Replacing the shape function ψ with the standard Lagrangian shape functions ψ_i, $\forall\, i \in \Sigma_h$, and applying again the integration by parts to the second member on the left-hand side of the previous equation, one obtains

$$\int_{\Omega_{e_i}}\psi_i\cdot\mathbf{f}(\mathsf{u}^h),\mathrm{d}\Omega + \int_{\partial\Omega_{e_i}\cap\partial\Omega}\psi_i(\mathbf{f}(\mathsf{u}^\partial) - \mathbf{f}(\mathsf{u}^h))\cdot\boldsymbol{n}\,\mathrm{d}\partial\Omega = 0,$$

where Ω_{e_i} is the union of the elements that have the DOF i in common. The boundary integral on the previous equation can be regarded as the contribution that must be added to the residuals computed on the domain elements without considering the boundary conditions. Note that the correction flux, $\mathbf{f}(\mathsf{u}^\partial) - \mathbf{f}(\mathsf{u}^h)$, becomes null as the boundary conditions are correctly enforced into the numerical scheme. With the Eq. (9) in mind, the complete RD space discretization, which includes the imposition of the boundary conditions, can be written as follows

$$\sum_{e\in\mathcal{E}_h^i}\Phi_i^e + \sum_{f\in\mathcal{F}_h^i}\Phi_i^{f,\partial} = 0, \quad \forall i \in \Sigma_h, \tag{19}$$

where the boundary residuals reads

$$\Phi_i^{f,\partial} = \int_{\partial\Omega_{e_i}\cap\partial\Omega} \psi_i(\mathbf{f}(\mathsf{u}^\partial) - \mathbf{f}(\mathsf{u}^h)) \cdot \boldsymbol{n}\, \mathrm{d}\partial\Omega$$

and $\mathcal{F}_{h,i}$ is the set of the boundary faces which share the DOF i; the set is empty if i is not on the boundary of the domain.

In the case of an a inviscid wall, the boundary conditions require that the normal component of the velocity to the plane is null: $\boldsymbol{v}\cdot\hat{\boldsymbol{n}} = 0$, with $\hat{\boldsymbol{n}}$ is the outward normal versor to the boundary face. The flux function and the correction flux read

$$\mathbf{f}^a(\mathsf{u}^\partial_{\text{wall}})\cdot\hat{\boldsymbol{n}} = \begin{pmatrix} 0 \\ P\hat{\boldsymbol{n}} \\ 0 \end{pmatrix}, \qquad \left(\mathbf{f}^a(\mathsf{u}^\partial_{\text{wall}}) - \mathbf{f}^a(\mathsf{u}^h)\right)\cdot\hat{\boldsymbol{n}} = -v_n \begin{pmatrix} \rho \\ \rho\boldsymbol{v} \\ E^t + P \end{pmatrix},$$

respectively, with $v_n = \boldsymbol{v}\cdot\hat{\boldsymbol{n}}$.

The inflow/outflow boundary conditions can be imposed specifying the state $\mathsf{u}^\partial_{\text{in/out}}$. As standard practice the advective flux function is linearized as

$$\mathbf{f}^a(\mathsf{u}^\partial_{\text{in/out}})\cdot\hat{\boldsymbol{n}} \simeq \mathsf{A}_n^+(\mathsf{u}^h) + \mathsf{A}_n^-(\mathsf{u}^\partial_{\text{in/out}}),$$

where $\mathsf{A}_n^\pm(\mathsf{u}) = \mathsf{R}_n(\mathsf{u})\,\Lambda_n^\pm(\mathsf{u})\,\mathsf{L}_n(\mathsf{u})$, The correction flux for the imposition of the inflow/outflow boundary conditions reads

$$\left(\mathbf{f}^a(\mathsf{u}^\partial_{\text{in/out}}) - \mathbf{f}^a(\mathsf{u}^h)\right)\cdot\hat{\boldsymbol{n}} = \mathsf{A}_n^-(\mathsf{u}^h)(\mathsf{u}^\partial_{\text{in/out}} - \mathsf{u}^h).$$

The same procedure used to impose the inflow/outflow boundary conditions can be used to specify the far-field boundary conditions.

For a viscous fluid, at the solid surface, the adiabatic and no-slip boundary condition is applied, in this case the velocity and normal heat flux on the wall must be zero: $\boldsymbol{v}|_{\text{wall}} = 0$, $\boldsymbol{q}\cdot\hat{\boldsymbol{n}}|_{\text{wall}} = 0$. In the RD scheme, the adiabatic no-slip boundary conditions are generally applied by initializing the velocity field such that the velocity components are zero on the wall and enforcing the residual of the momentum equation to be also zero on the boundary nodes at each iteration, this corresponds to impose strongly the boundary conditions for the momentum equation. Nothing is explicitly done for the residual associated to the energy equation, this is partially justified by the fact that on the wall

$$\int_{\partial\Omega_{e_i}\cap\partial\Omega} (\boldsymbol{v}\cdot\mathbb{S} - \boldsymbol{q}) = -\int_{\partial\Omega_{e_i}\cap\partial\Omega} \boldsymbol{q},$$

because the velocity is zero, so if the previous integral is not computed, this is equivalent to assume that $\boldsymbol{q}$ is zero on the wall. However this consideration is not true in practice, and imposing the boundary conditions in such a way does not guarantee that the heat flux is zero on the wall.

An effective way to impose the boundary conditions for the energy equation makes use of the correction flux technique showed before. Since the velocity is zero on the wall, the viscous flux function in the direction normal to the wall becomes

$$\mathbf{f}^v(\mathsf{u})\cdot\hat{\boldsymbol{n}} = \begin{pmatrix} 0 \\ \mathbb{S}\cdot\hat{\boldsymbol{n}} \\ \kappa\boldsymbol{\nabla}T\cdot\hat{\boldsymbol{n}} \end{pmatrix},$$

consequently the correction flux on the wall is

$$\mathbf{f}^v(\mathsf{u}^\partial_{\text{wall}}) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\kappa\boldsymbol{\nabla}T\cdot\boldsymbol{n} \end{pmatrix}.$$

The gradient of the temperature for a perfect gas can be easily expressed as function of the conservative variables and their gradient, for example, considering explicitly the x component for simplicity

$$\begin{aligned}\frac{\partial T}{\partial x} &= \frac{\gamma-1}{\mathcal{R}}\frac{\partial}{\partial x}\left(\frac{E^t}{\rho}-\frac{1}{2}\|\boldsymbol{v}\|^2\right)\\ &= \frac{\gamma-1}{\mathcal{R}}\left[\frac{\partial}{\partial x}\left(\frac{E^t}{\rho}\right)-\frac{1}{2}\left(\frac{\partial}{\partial x}\left(\frac{m_x^2}{\rho^2}\right)+\frac{\partial}{\partial x}\left(\frac{m_y^2}{\rho^2}\right)\right)\right]\\ &= \frac{\gamma-1}{\mathcal{R}}\left[\frac{1}{\rho}\frac{\partial E^t}{\partial x}-\frac{E^t}{\rho^2}\frac{\partial \rho}{\partial x}-\underbrace{\left(\frac{m_x}{\rho^2}\frac{\partial m_x}{\partial x}-\frac{m_x^2}{\rho^3}\frac{\partial \rho}{\partial x}+\frac{m_y}{\rho^2}\frac{\partial m_y}{\partial x}-\frac{m_y^2}{\rho^3}\frac{\partial \rho}{\partial x}\right)}_{=0,\text{wall}}\right],\end{aligned}$$

and therefore

$$\boldsymbol{\nabla} T = \frac{\gamma-1}{\mathcal{R}}\left(\boldsymbol{\nabla} E^t - \frac{E^t}{\rho}\boldsymbol{\nabla}\rho\right)\frac{1}{\rho}.$$

The reconstructed gradients of the conservatives variables are used in the previous equation. As usual the residual of the momentum equation is explicitly imposed to be zero in order to apply the strong boundary conditions for the velocity on the wall.

4. Gradient recovery strategies

As explained in Section 3, one has to assume that a continuous value of the gradient of the numerical solution is available on the faces of the elements. The strategy adopted here to achieve this goal consists in recovering the gradients at every DOF of the element, then the nodal values of gradients are interpolated with the Lagrangian shape functions. It is evident that the gradient reconstruction represents a key point to get a higher order of accuracy; on the other hand the computational effort in the gradient reconstruction should be as limited as possible.

In [3] different reconstruction strategies were considered, and it was found that the so-called Super-convergent Patch Recovery method proposed by Zienkiewicz and Zhu [38, 39] (SPR-ZZ) offers the possibility to reconstruct the gradients with the same order of accuracy of the solution, without spoiling the compactness of the numerical scheme. For sake of clarity, this method is briefly recalled here.

4.1. Super-convergent Patch Recovery

In the field of the Finite Element Method applied to the mechanical structures, it is known that if the gradient of the numerical solution is sampled at certain points on the element, it may have the same order of accuracy of the solution [40]. It can be shown that in the case of a segment element such particular points correspond to the Gauss-Legendre points [16]. By tensor product, such points can be defined also for quadrangles and hexahedrons. For triangles or tetrahedrons such property cannot be rigorously shown, but numerical experiments confirm it.

Using this fact, it is possible to compute a high-order accurate gradient within the whole element. Indeed, if at the super-convergent points, the value of the gradient is accurate to order $k+1$, by using a polynomial of degree k (the same of the solution), it is possible to obtain a high-order approximation everywhere within the element if this polynomial fits the super-convergent gradient values in a least square manner within a patch of elements for which the number of super-convergent points is greater than the number of parameters in the polynomial. Such a technique is called super-convergent patch recovery, introduced by Zienkiewicz and Zhu (SPR-ZZ) [38, 39].

Consider, for simplicity, a two dimensional scalar problem and assume that u^h is the piecewise continuous polynomial interpolation of the solution of degree k. The aim is to reconstruct the gradient of the numerical solution at the DOFs, with $(k+1)$-th order of accuracy. For each vertex of the grid a patch is defined as the union of all the elements that share that vertex, see Fig. 1. On the patch, the components of the reconstructed gradient are written in a polynomial form as follows

$$\widetilde{\frac{\partial u^h}{\partial x}} = \boldsymbol{p}^{\mathrm{T}}\boldsymbol{a}_x, \quad \text{and} \quad \widetilde{\frac{\partial u^h}{\partial y}} = \boldsymbol{p}^{\mathrm{T}}\boldsymbol{a}_y, \tag{20}$$

with

$$\boldsymbol{p}^{\mathrm{T}}(\boldsymbol{x}) = (1, x, y, x^2, \ldots, x^k, x^{k-1}y, \ldots, y^k),$$

$$\boldsymbol{a}_x = (a_{x_1}, a_{x_2}, \ldots, a_{x_m}) \quad \text{and} \quad \boldsymbol{a}_y = (a_{y_1}, a_{y_2}, \ldots, a_{y_m}).$$

On each element of the patch, the gradient of the numerical solution has the highest accuracy at the super-convergent points, thus the high-order accurate expression of the reconstructed gradient on the patch is obtained by a least square fitting of the polynomial expressions (20) to the values of the gradient at the super-convergent points of the patch.

Assuming that N_{s_i} super-convergent points, $\boldsymbol{x}_{i,\ell}, \ell = 1 \ldots N_{s_i}$, are available over the patch built around the vertex i, the values of the coefficients $\boldsymbol{a}_{x_i}$ and $\boldsymbol{a}_{y_i}$ are obtained by introducing the following functions

$$F_{x_i} = \sum_{k=1}^{N_{s_i}} \left(\frac{\partial u^h}{\partial x}(\boldsymbol{x}_{i,k}) - \boldsymbol{p}_{i,k}^{\mathrm{T}} \boldsymbol{a}_{x_i} \right)^2 \quad \text{and} \quad F_{y_i} = \sum_{k=1}^{N_{s_i}} \left(\frac{\partial u^h}{\partial y}(\boldsymbol{x}_{i,k}) - \boldsymbol{p}_{i,k}^{\mathrm{T}} \boldsymbol{a}_{y_i} \right)^2,$$

with $\boldsymbol{p}_{i,k} = \boldsymbol{p}(\boldsymbol{x}_{i,k})$, and by minimizing them with respect to the polynomial coefficients, that is

$$\frac{\partial F_{x_i}}{\partial \boldsymbol{a}_{x_i}} = 0 \quad \text{and} \quad \frac{\partial F_{y_i}}{\partial \boldsymbol{a}_{y_i}} = 0.$$

The minimization problem for the patch around the vertex i requires the solution of the following linear systems

$$A_i \boldsymbol{a}_{x_i} = \boldsymbol{b}_{x_i}^h \quad \text{and} \quad A_i \boldsymbol{a}_{y_i} = \boldsymbol{b}_{y_i}^h, \tag{21}$$

in a least-square sense, where

$$\boldsymbol{b}_{x_i}^h = \begin{pmatrix} \dfrac{\partial u^h}{\partial x}(\boldsymbol{x}_{i,1}) \\ \dfrac{\partial u^h}{\partial x}(\boldsymbol{x}_{i,2}) \\ \vdots \\ \dfrac{\partial u^h}{\partial x}(\boldsymbol{x}_{i,N_{s_i}}) \end{pmatrix}, \quad \boldsymbol{b}_{y_i}^h = \begin{pmatrix} \dfrac{\partial u^h}{\partial y}(\boldsymbol{x}_{i,1}) \\ \dfrac{\partial u^h}{\partial y}(\boldsymbol{x}_{i,2}) \\ \vdots \\ \dfrac{\partial u^h}{\partial y}(\boldsymbol{x}_{i,N_{s_i}}) \end{pmatrix} \quad \text{and} \quad A_i = \begin{pmatrix} 1 & x_{i,1} & y_{i,1} & \ldots & y_{i,1}^k \\ 1 & x_{i,2} & y_{i,2} & \ldots & y_{i,2}^k \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{i,N_{s_i}} & y_{i,N_{s_i}} & \ldots & y_{i,N_{s_i}}^k \end{pmatrix}.$$

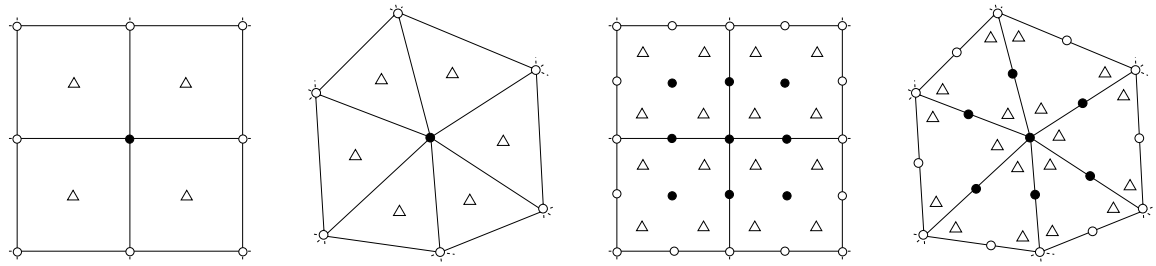
Once the vectors $\boldsymbol{a}_{x_i}$ and $\boldsymbol{a}_{y_i}$ are known, the reconstructed gradient $\widetilde{\boldsymbol{\nabla} u^h}$ can be evaluated at every point within the patch around the vertex i.

The dimensions of the matrix A_i are determined by the number of super-convergent points N_{s_i} and by the degree of the polynomials used to express the reconstructed gradient, that is $A_i \in \mathbb{R}^{N_{s_i} \times m}$, where m is the number of the coefficients in the vectors $\boldsymbol{a}_{x_i}$ and $\boldsymbol{a}_{y_i}$. The problem admits an unique solution if $\operatorname{Rank} A_i = m$, which is always satisfied in the case in which $N_{s_i} \geq m$. Note that the SPR-ZZ method requires only the solution of small (and local) linear systems instead of a single large system as that obtained by a global L^2 projection.

Generally, the number of elements contained in the patch is such that the condition $N_{s_i} \geq m$ is always satisfied, this means that the gradient reconstruction is compact because it involves only the elements contained within the support of a grid node.

It is important to remark that the SPR-ZZ method solves for the vector of the polynomial coefficients, $\boldsymbol{a}_x$ and $\boldsymbol{a}_y$ over each patch, instead of solving for the nodal values of the reconstructed gradient as in the classical least-square gradient reconstruction (see Appendix A). Once the polynomial coefficients are computed on the patch associated to a grid vertex, they can be used to evaluate the gradient in any point on that patch. As matter of fact, for the extra nodes introduced by the higher approximation of the solution, the gradient is reconstructed by simply evaluating on the patch, at the coordinates of the nodes, the polynomial function constructed for the nearest grid vertex. Note that these nodes may belong to more than one patch, each of them is equally valid to evaluate the gradient. An arithmetic average is used to uniquely define the reconstructed gradient at these nodes.

The same idea is used to get the value of the reconstructed gradient at the boundary of the domain. In fact, for the nodes belonging to the boundary, the condition $N_{s_i} \geq m$ might not be satisfied without enlarging the stencil, otherwise



(a) (b) (c) (d)

Figure 1: Interior super-convergent patches for quadrilateral and triangular elements: (a-b) linear elements, (c-d) quadratic elements. The symbols (○) indicate the patch assembly points, the symbols (●) indicate the points where the gradient is reconstructed and the symbols (△) indicate the super-convergent points.

the problem is ill conditioned. To avoid the use of larger stencils for boundary nodes, it is possible to obtain the value of the reconstructed gradient at the boundary by evaluating, at the boundary node, the polynomial expansion already computed for the nearest domain vertex, *i.e.*, the interior patch is used to evaluate the gradient at the boundary.

Another advantage of the SPR-ZZ approach is that being based on an element-wise evaluation (the super-convergent points inside the element of the patch where the gradient is evaluated), the stencil of the SPR-ZZ method is smaller than that required by the direct used of the least-square gradient reconstruction method, which is based on a nodal evaluation of the solution. In fact, it is shown here that if the stencil used for the gradient reconstruction is made only by the nearest neighbor elements, the least square gradient reconstruction is at the most second order accurate.

For a quadrangle, the super-convergent points are defined uniquely. Considering a reference segment defined as $x = [-1, 1]$, the super-convergent point is the point $x = 0$ in the case of a linear element, while in the case of a quadratic element the super-convergent points have coordinates $\pm 1/\sqrt{3}$. The super-convergent points on the reference quadrangle are simply obtained by a tensorial product of the points defined on the reference segment.

For a linear triangle, the super-convergent point is the point with barycentric coordinated $\boldsymbol{\lambda} = (1/3, 1/3, 1/3)$, while in the case of a quadratic triangle the super-convergent points are not unique, different choices are available. In Fig. 2 are shown three examples of super-convergent points used. In the first option (Fig. 2-(a)) are used three points with barycentric coordinates

$$\boldsymbol{\lambda}_1 = \left(\frac{2}{3}, \frac{1}{3}, \frac{1}{3}\right), \quad \boldsymbol{\lambda}_2 = \left(\frac{1}{3}, \frac{2}{3}, \frac{1}{3}\right), \quad \text{and} \quad \boldsymbol{\lambda}_3 = \left(\frac{1}{3}, \frac{1}{3}, \frac{2}{3}\right).$$

In the second option (Figure 2-(b)) four points are used with barycentric coordinates

$$\boldsymbol{\lambda}_1 = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right), \quad \boldsymbol{\lambda}_2 = (0.6, 0.2, 0.2), \quad \boldsymbol{\lambda}_3 = (0.2, 0.6, 0.2), \quad \boldsymbol{\lambda}_4 = (0.2, 0.2, 0.6).$$

Another option (Fig. 2-(c)) consists in taking as super-convergent points the three points with barycentric coordinates

$$\boldsymbol{\lambda}_1 = \left(\frac{1}{2}, \frac{1}{2}, 0\right), \quad \boldsymbol{\lambda}_2 = \left(0, \frac{1}{2}, \frac{1}{2}\right), \quad \boldsymbol{\lambda}_3 = \left(\frac{1}{2}, \frac{1}{2}, 0\right).$$

4.2. Results and discussion

To study the accuracy of the presented gradient reconstruction strategy, the following function is used

$$u = -\cos(2\pi\eta) \exp\left\{\frac{\xi\left(1 - \sqrt{1 + 16\pi^2\nu^2}\right)}{2\nu}\right\}, \tag{22}$$

with $\eta = a_y x - a_x y$ and $\xi = a_x x + a_y$. Here $a_x = 0.5$, $a_y = \sqrt{3}/2$ and $\nu = 0.01$. The solution, shown in Fig. 3, is infinitely differentiable with continuous gradient. The computations of the reconstructed gradients are performed on

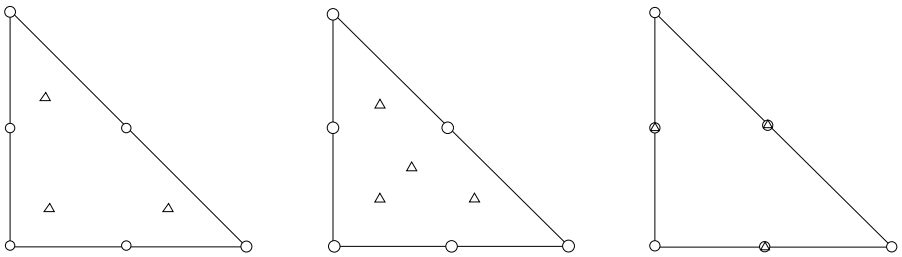


Figure 2: Three different examples of super-convergent points for quadratic triangles.

four different kind of grids, shown in Fig. 4, namely unstructured grids of triangles, quadrangles and hybrid elements and highly deformed unstructured meshes of triangles, obtained randomly perturbing a regular grid.

The SPR-ZZ technique is compared against the weighted-area (hereafter called also Green-Gauss), L^2-projection and least-square procedures. Note that in the least-square method, for each node, only the direct neighbor nodes are considered, to avoid a larger computational stencil.

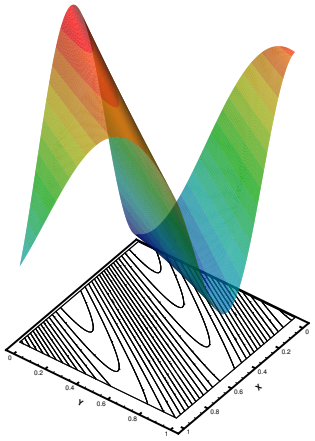
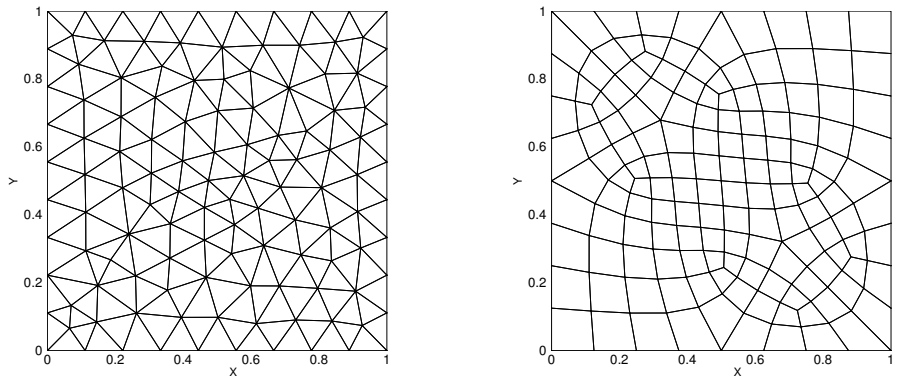


Figure 3: Graphical representation of the Eq. (22) used to test the gradient recovery methods.

The error of the reconstruction procedure is computed as the L^2 norm of the difference between the computed gradient, $\widetilde{\boldsymbol{\nabla} u^h}$, and the exact gradient, $\boldsymbol{\nabla} u_{\text{ex}}$, for each spatial component

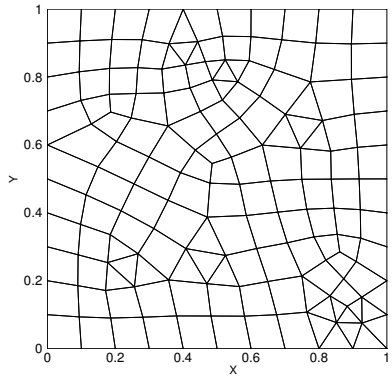
$$\epsilon_{L^2} = \sqrt{\frac{\int_\Omega \left(\widetilde{\boldsymbol{\nabla} u^h} - \boldsymbol{\nabla} u^h_{\text{ex}}\right)^2 \mathrm{d}\Omega}{\int_\Omega \left(\boldsymbol{\nabla} u_{\text{ex}}\right)^2 \mathrm{d}\Omega}}.$$

In Fig. 5 are shown the L^2 errors of different gradient reconstruction methods with triangular grids. In the case of linear elements, the differences between the reconstructed methods are small and all the schemes reach almost the second order accuracy. The L^2-Projection and the SPR-ZZ methods have the smallest level of error, but the former scheme is much more expensive because it requires the solution of a global linear system. In the case of quadratic elements, the methods have an order of accuracy no more than two, except for the SPR-ZZ method which shows third order accuracy. It is also worth noticing that the errors obtained with the SPR-ZZ method are one order of magnitude smaller than those obtained with other methods. In Fig. 6 are reported the errors obtained with the SPR-ZZ procedure

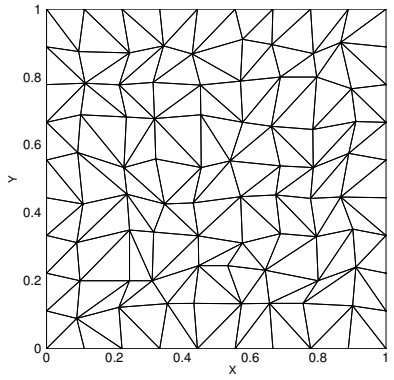


(a)

(b)



(c)



(d)

Figure 4: Example of different kinds of grid used to test the accuracy of the gradient recovery procedures.

on quadratic triangular elements, for the three different sampling strategies shown in Fig. 2. It is evident that the first strategy guaranties the smallest level of error, while the four-points strategy does not introduce any improvement.

The errors of the recovery methods on unstructured grids of quadrangles and of hybrid elements are reported in Fig. 7 and Fig. 8, respectively. The behavior of the recovery methods is the same observed in the case of triangular grids. Fig. 9 shows the errors computed on a sequence of highly distorted triangular grids, the performance of the recovery methods is not optimal anymore due to very poor quality of the meshes, nevertheless the errors obtained with the SPR-ZZ methods are always much smaller than those obtained with other procedures.

5. Implicit time integration

The numerical solution of the RD method is obtained by solving the non-linear system of equations (19), with an approximated Newton's method of this form

$$\left[\frac{\mathbb{I}}{\Delta t^n} + J(\mathsf{u}^n)\right]\Delta\mathsf{u}^n = -\mathbf{R}(\mathsf{u}^n)$$

$$\mathsf{u}^{n+1} = \mathsf{u}^n + \Delta\mathsf{u}^n,$$

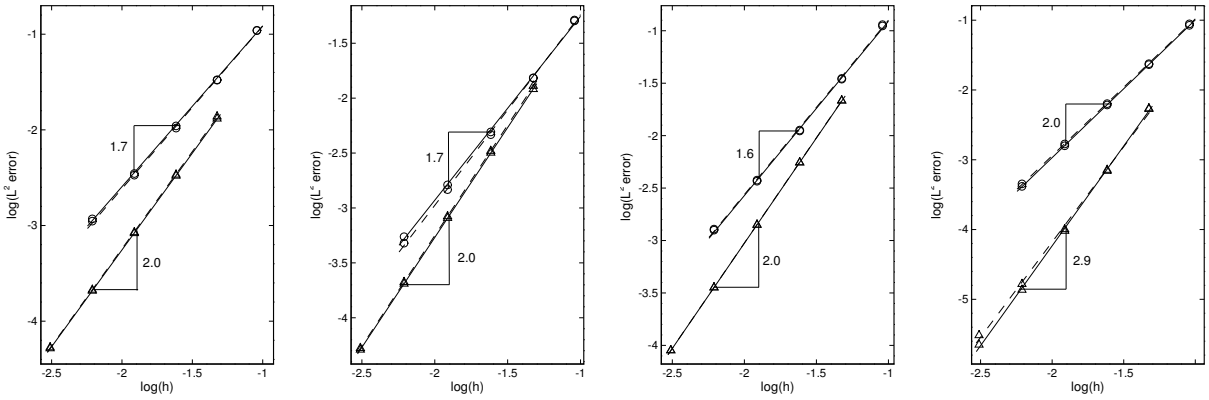


Figure 5: L^2 errors of different gradient recovery methods with linear (∘) and quadratic (△) elements on an unstructured grids of triangles. Solid and dashed lines represent the error respectively of the x and y components of the gradients. From left to right: Weighted area, L^2-Projection, Least square and SPR-ZZ methods. The mean slopes of curves are also reported also and $h = 1/\sqrt{N_{\text{dof}}}$.

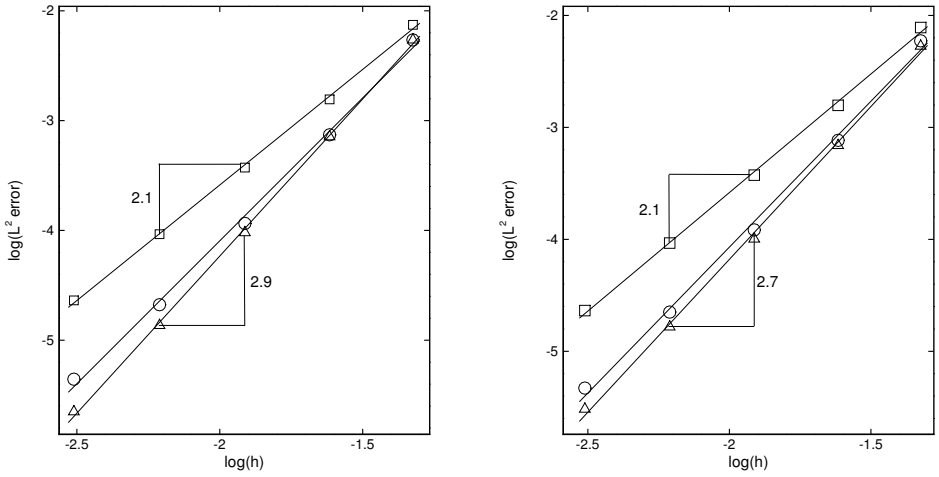


Figure 6: L^2 errors of the SPR-ZZ method for quadratic triangular elements with different sampling strategies: (△) strategy Fig. 2-(a), (∘) strategy Fig. 2-(b) and (□) strategy Fig. 2-(c). On the left errors for the x component of the gradient, on the right errors for the y component of the gradient and $h = 1/\sqrt{N_{\text{dof}}}$.

where Δt^n is time step used to dampen the solution, with $\Delta t^n \to \infty$ as $n \to \infty$. The evolution of the time step is controlled by the CFL number, that is chosen according to the following law [35]

$$\text{CFL}^n = \text{CFL}^{n-1} \frac{\|\mathbf{R}(\mathsf{u}^{n-2})\|_\infty}{\|\mathbf{R}(\mathsf{u}^{n-1})\|_\infty},$$

starting from a low CFL number. The iterative process is stopped when the residual of the equations becomes small enough respect to the initial residual.

The Newton's method creates a linear system which is solved at each time step with a GMRES method [30]. The linear system requires to compute the Jacobian of $\mathbf{R}$. Unfortunately, explicit computation of the Jacobian matrix resulting from the linearization of the high order residuals is extremely expensive, if not impossible. Since the GMRES

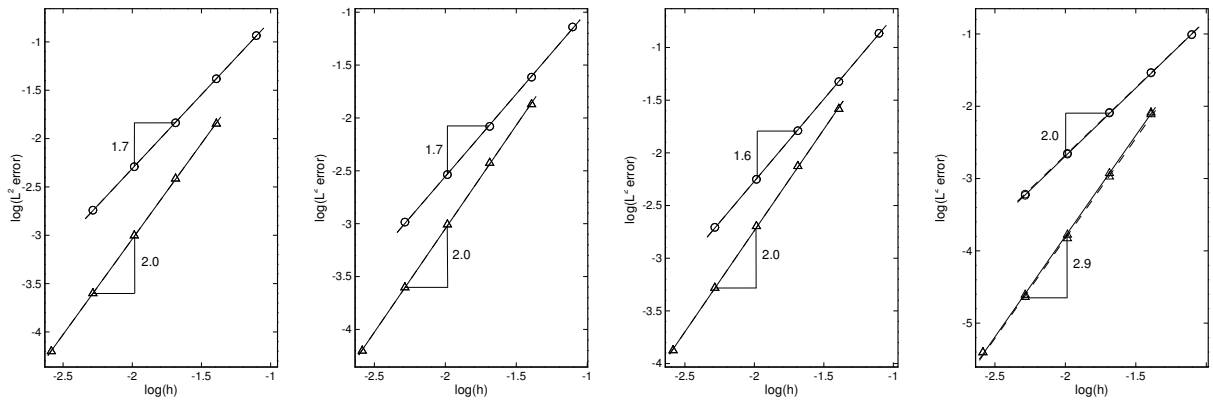


Figure 7: L^2 errors of different gradient recovery methods with linear (∘) and quadratic (△) elements on an unstructured grids of quadrangles. Solid and dashed lines represent the error respectively of the x and y components of the gradients. From left to right: Weighted area, L^2-Projection, Least square and SPR-ZZ methods. The mean slopes of curves are also reported also and $h = 1/\sqrt{N_{\text{dof}}}$.

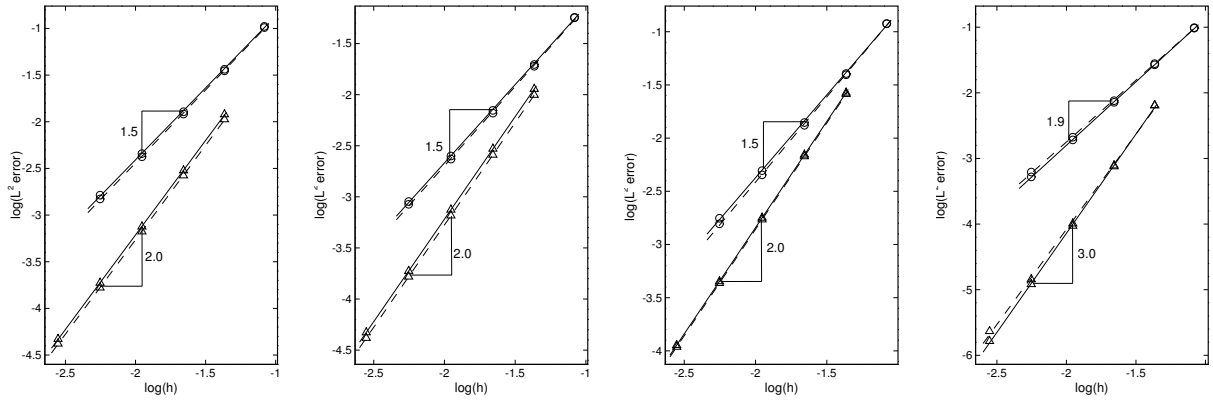


Figure 8: L^2 errors of different gradient recovery methods with linear (∘) and quadratic (△) elements on an unstructured grids of hybrid elements. Solid and dashed lines represent the error respectively of the x and y components of the gradients. From left to right: Weighted area, L^2-Projection, Least square and SPR-ZZ methods. The mean slopes of curves are also reported also and $h = 1/\sqrt{N_{\text{dof}}}$.

algorithm only requires the matrix of the linear system in matrix-vector products, the following approximation is used [34, 9, 19, 7]

$$\left(\frac{\mathbb{I}}{\Delta t^n} + J(\mathsf{u}^n)\right)\Delta\mathsf{u} \simeq \frac{\mathbb{I}}{\Delta t^n}\Delta\mathsf{u} + \frac{\mathbf{R}(\mathsf{u}^n + \epsilon\Delta\mathsf{u}) - \mathbf{R}(\mathsf{u}^n)}{\epsilon},$$

with ϵ a small step size chosen as [26]

$$\epsilon = \frac{\sqrt{1 + \|\mathsf{u}^n\|_{L_2}}}{\|\Delta\mathsf{u}\|_{L_2}}\,\epsilon_{\text{rel}}, \qquad \epsilon_{\text{rel}} = 10^{-8}.$$

To accelerate the convergence of the iterative linear solver, right preconditioning is used. In this work the LU-SGS [18] preconditioning is considered, which reads

$$P = (L + D)D^{-1}(D + U),$$

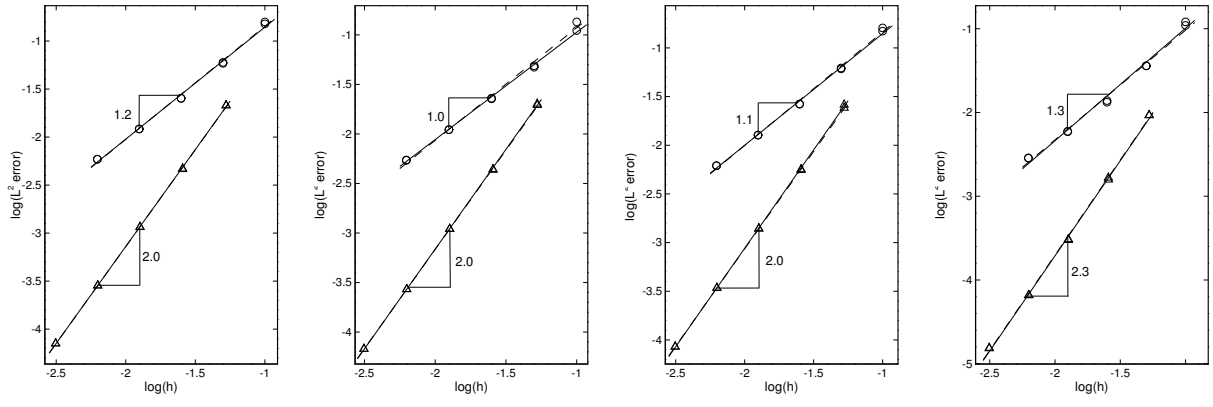


Figure 9: L^2 errors of different gradient recovery methods with linear (∘) and quadratic (△) elements on an unstructured grids of randomly distorted triangles. Solid and dashed lines represent the error respectively of the x and y components of the gradients. From left to right: Weighted area, L^2-Projection, Least square and SPR-ZZ methods. The mean slopes of curves are also reported also and $h = 1/\sqrt{N_{\text{dof}}}$.

where D, is the diagonal block of the approximated Jacobian, L and U are the lower and the upper parts of the Jacobian, respectively.

6. Numerical results

In this section we use the proposed numerical scheme to discretize the compressible Navier-Stokes equations. Although the theoretical framework has been presented for a generic discretization order and type of element, here we limit ourselves to simulations with linear and quadratic approximation of the solution on grids made of triangles (2D) or tetrahedra (3D), for practical reasons only.

6.1. Manufactured solutions

Up to the best of our knowledge, there is no known exact solution to the Navier Stokes equations. In order to study the accuracy properties of the method we propose, we rely on the manufactured solutions. This is a flexible way to create analytical solutions for problems of interest, in this way it is possible to verify rigorously the accuracy of the numerical scheme. Whi this approach, the form of the solution is chosen a priori, and the differential operator of the governing equation is applied to the chosen solution. The reminder of this operation is then used as source term for the original governing equation.

Although the form of the solution is arbitrary, it should be smooth and infinitely differentiable. Here, the form of the primitive variables is taken as a function of sines and cosines [29]

$$\rho(x,y) = \rho_0 + \rho_x \sin\left(\frac{a_x \pi x}{L}\right) + \rho_y \cos\left(\frac{a_y \pi y}{L}\right) + \rho_{xy} \cos\left(\frac{a_{xy} \pi xy}{L^2}\right)$$

$$u(x,y) = u_0 + u_x \sin\left(\frac{a_x \pi x}{L}\right) + u_y \cos\left(\frac{a_y \pi y}{L}\right) + u_{xy} \cos\left(\frac{a_{xy} \pi xy}{L^2}\right)$$

$$v(x,y) = v_0 + v_x \cos\left(\frac{a_x \pi x}{L}\right) + v_y \sin\left(\frac{a_y \pi y}{L}\right) + v_{xy} \cos\left(\frac{a_{xy} \pi xy}{L^2}\right)$$

$$P(x,y) = P_0 + P_x \cos\left(\frac{a_x \pi x}{L}\right) + P_y \sin\left(\frac{a_y \pi y}{L}\right) + P_{xy} \sin\left(\frac{a_{xy} \pi xy}{L^2}\right)$$

for the density, the velocity components and the pressure, and where L is the length of the edge of the domain, assumed to be a square. In Table. 1 are reported the constants used in the previous expressions The conservative variables

Variable ϕ	ϕ_0	ϕ_x	ϕ_y	ϕ_{xy}	a_{ϕ_x}	a_{ϕ_y}	$a_{\phi_{xy}}$
$\rho(Kg/m^3)$	1	0.1	-0.1	0.08	0.75	1.0	1.25
$u(m/s)$	70	4	-12	7	5/3	1.5	0.6
$v(m/s)$	90	-20	4	-11	51.5	1.0	0.9
$P(N/m^2)$	1×10^5	-0.3×10^5	0.2×10^5	-0.25×10^5	51.5	1.25	0.75

Table 1: Values of the constants used in the manufactured solutions method.

are first computed from the previous definition of the primitive variables and then are made dimensionless using as reference quantities the values at the point of coordinates $(0, 0)$, while the spatial coordinates are made dimensionless with L. In figure 10 are reported the contours of the conservative variables used as manufactured solution.

Dirichlet boundary conditions are applied on the boundaries of the domain, and starting from an uniform initialization, the solution is made evolve until the norm of the residual of the conservative variables is dropped by ten orders of magnitude. The discretization error is taken as the normalized L^2 norm of the difference between the numerical solution and the exact one, namely

$$\epsilon_{L^2} = \sqrt{\frac{\int_\Omega \|\mathsf{u}^h - \mathsf{u}_{\text{ex}}\|^2 \, \mathrm{d}\Omega}{\int_\Omega \|\mathsf{u}_{\text{ex}}\|^2 \, \mathrm{d}\Omega}},$$

and the accuracy study is performed on sequence of four unstructured grids, of the same type shown in Fig. 4-(a), made of 228, 898, 3 588 and 14 412 triangles. The aim of the present study, is not only to verify the formal order of accuracy of the numerical scheme, but also to investigate if the observed order is kept constant when the relative importance of advection and diffusion changes. To this purpose, the observed order of accuracy is computed for different Reynold numbers.

Two different types of gradient reconstruction are considered, the SPR-ZZ and the Green-Gauss methods, because it has been already observed in the study performed for a scalar function, that there is only a limited difference between the Green-Gauss method and other more sophisticated methods.

In Fig. 11, are reported, for different Reynolds numbers, the observed orders of accuracy, for the numerical solution and the reconstructed gradient, with the linear and non-linear schemes and the two type of gradient reconstruction. The observed order of accuracy is taken as the mean slope, computed with a linear least square interpolation, of the variables $\log(1/\sqrt{N_{\text{dof}}})$ and $\log(\epsilon_{L^2})$. Note that only the results for the x component of the gradients are reported, results for the y component are very similar.

Consider first the linear scheme. At the second order there is almost no difference between the results obtained with the two types of gradient reconstructions in the observed order of accuracy of the solution: both methods guarantee formal second order accuracy for the solution. However only the SPR-ZZ method gives also second order accuracy for the gradient. With quadratic elements, the situation changes drastically, only the SPR-ZZ method gives third order accuracy on the whole range of Reynolds numbers, for both solution and gradient. It is interesting to observe that in the diffusion limit, the use of a higher order reconstruction method for the gradients gives an increment of accuracy for the solution respect to the theoretical third order; eventually the theoretical order is reached in the advection dominated regime. The same behavior was also observed for scalar advection-diffusion problems. With the Green-Gauss method, the formal accuracy of the solution is lost in the diffusion dominated regime, and is recovered only when advective effects dominate the diffusive ones. It is worth noting also that the Green-Gauss method gives at most second order accuracy for the gradient, to be compared with the third order accuracy guaranteed by the SPR-ZZ method.

The same considerations done above for the linear scheme are also valid for the non-linear scheme, however in this case the loss of accuracy in the diffusion regime is more severe if gradients are not reconstructed properly, furthermore the loss of accuracy affects also the solution computed with linear elements.

6.2. Laminar flow over a NACA-0012 airfoil

In this test, a subsonic viscous flow over a NACA-0012 airfoil at zero angle of attack is considered, the free stream Mach number is 0.5 and Reynolds number is 5 000. This is a widely used test case for two dimensional laminar flows;

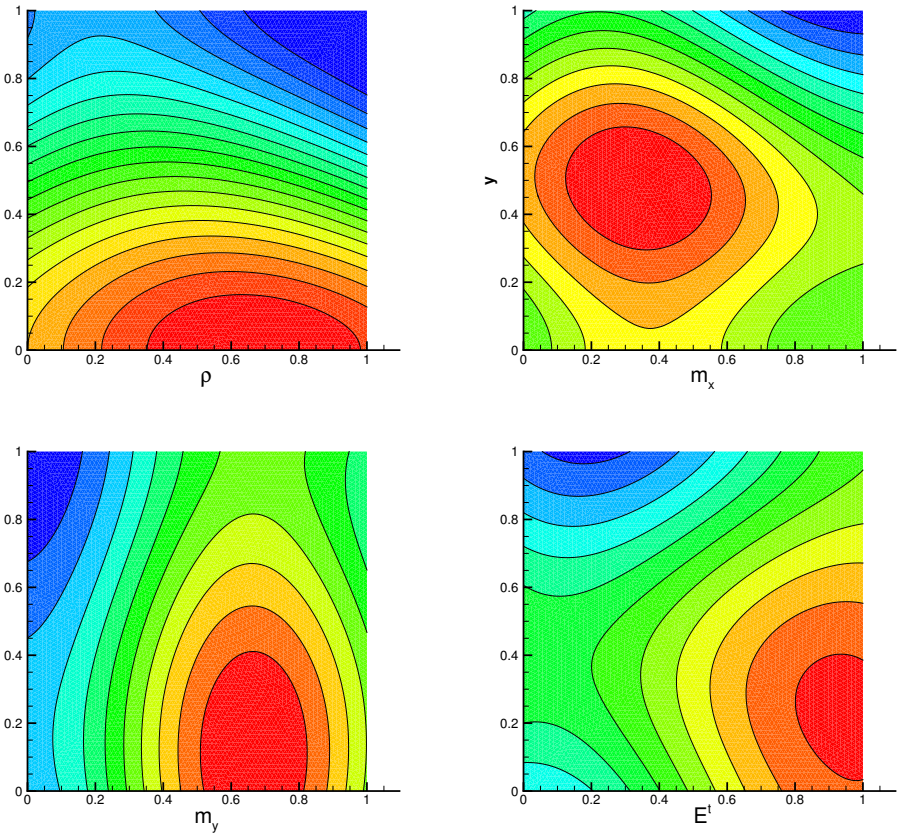


Figure 10: Form of the conservative variables used for the manufactured solution method.

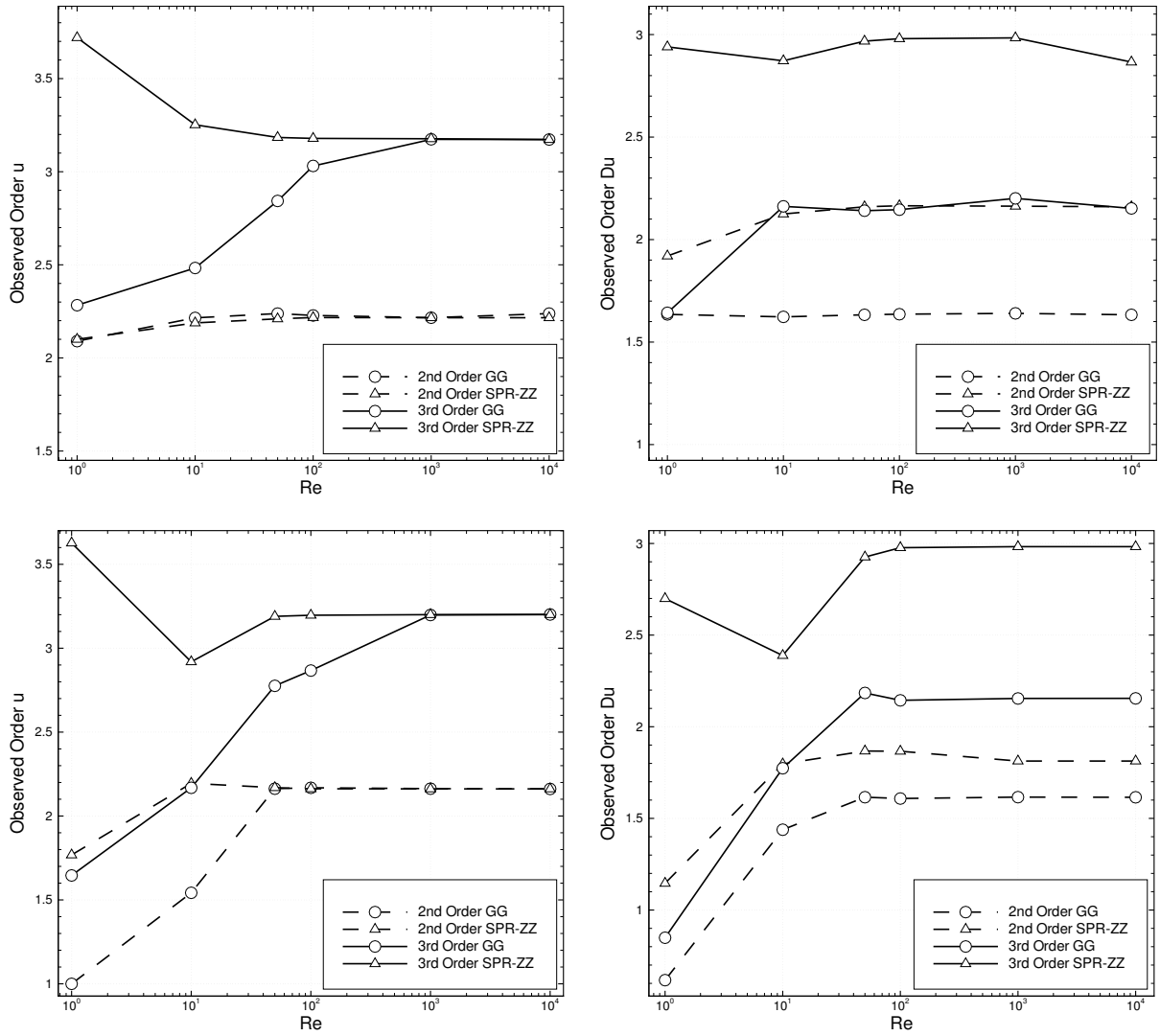


Figure 11: Observed orders of accuracy for the solution (left) and the gradient (right) in the manufactured solution test, at different Reynold numbers and with two type of gradient reconstruction. On the top row are reported the results for the linear scheme, on the bottom row are reported the results for the non-linear scheme.

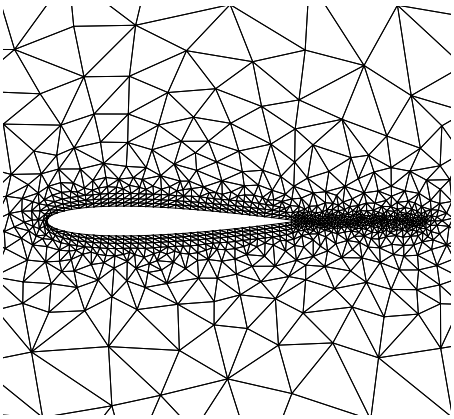


Figure 12: An example of computational grid used for the NACA-0012 test case.

a distinctive feature of this test case is a steady separation bubble near the trailing edge of the airfoil. An example of computational grid is displayed in Fig. 12, and three levels of grid refinement are considered in the simulations. The grid extends about 50 chords away from the airfoil. The airfoil boundary is considered adiabatic, no-slip and is represented by piece-wise quadratic elements, the far-field boundary condition is applied on the outer boundary of the domain. In all the simulation the steady state is considered to be reached when the L^2 norm of the density residual is drop by ten orders of magnitudes respect to the initial value.

In Fig. 13 are depicted the solutions computed with the linear scheme and the SPR-ZZ gradient reconstruction, for P1 and P2 elements. The solution with the P1 elements has been computed on a grid obtained from that with P2 elements (4 216 elements) and splitting each P2 triangle with four P1 triangles, in such a way the number of DOFs for the second and third order simulation is exactly the same. Note, in Fig. 13, that although there is not much difference in the Mach number contours between the second and the third order simulations, the streamlines near the trailing edge are very different, and only the third order scheme is able to reproduce the symmetric recirculation bubble. For the same simulations, in Fig. 14 and Fig. 15 are reported the pressure and skin friction coefficients profiles, respectively. Note the more regularity of the solution of the third order simulation respect to the second order one, for the same number of DOFs.

In order to compare the effect of the gradient reconstruction on the quality of the solution, in Fig. 16 is reported the absolute values of the lift coefficient as function of the number of DOFs, for linear and quadratic elements, with the SPR-ZZ and the Green-Gauss gradient reconstruction. Since the airfoil is symmetric and the angle of attack is zero, the theoretical value of the lift coefficient is zero; note that the mesh has no particular symmetries which could cancel out numerical errors. It is evident that, for the same number of DOF, the RD scheme with quadratic elements gives a better approximation of the problem, in particular the use the SPR-ZZ guarantees a great improvement in the numerical discretization. If a proper gradient reconstruction is not used, the gain in using a higher order RD scheme remains marginal respect to a formal second order approximation of the solution. For sake of completeness, in Table. 2 are also reported the value of pressure and viscous contributions to the drag coefficient. The comparison between the present results and those obtained with a Spectral Difference (SD) method and FV schemes are also summarized in Table. 2. A good agreement is obtained, and differences in the last decimal digits are most likely due to the effect of distance of the far-field boundary.

In addition, in Fig. 17 are reported the drag coefficient errors as function of the number of the DOFs and the CPU time required to perform the simulations. The error is computed as the absolute value of the difference between the drag coefficients reported in Table. 2 and a reference value computed on a finer grid with quadratic elements ($N_{\text{dof}} = 36\,044$) and the SPR-ZZ gradient reconstruction method. The benefit of using high order elements is evident.

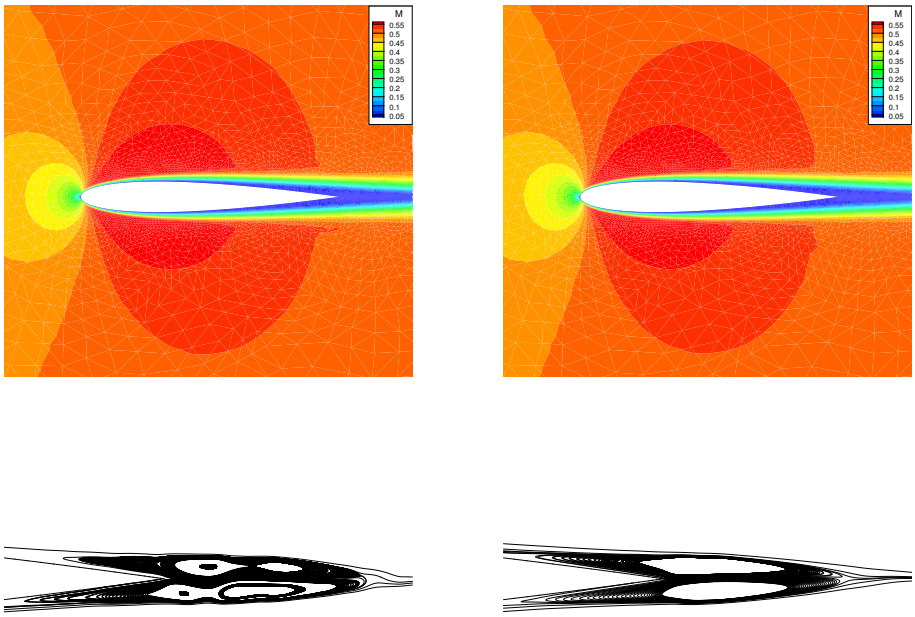


Figure 13: Mach Number contours (top) and streamlines near the trailing edge (bottom) for the second (left) and third (right) order linear scheme.

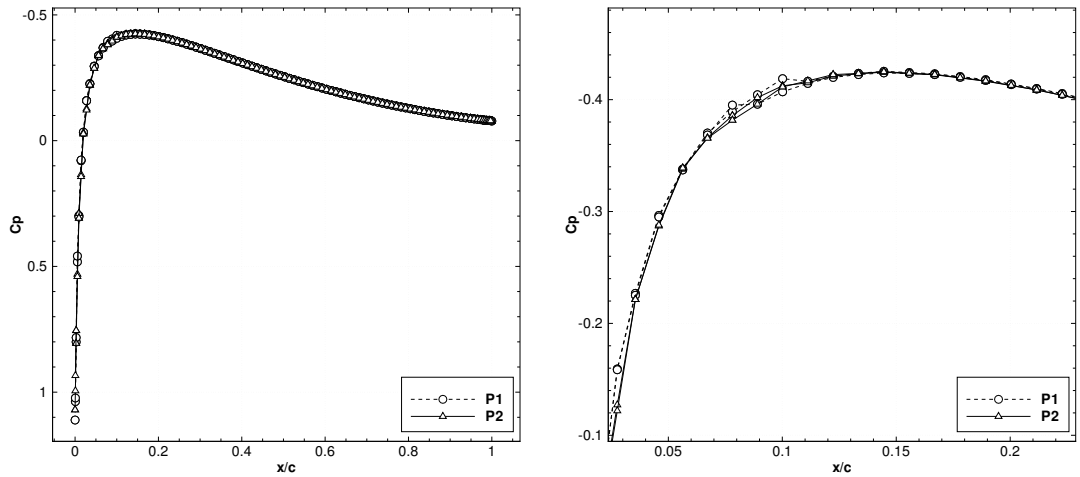


Figure 14: Pressure coefficient along the whole NACA-0012 airfoil for the second and third order simulations, with the same number of DOFs.

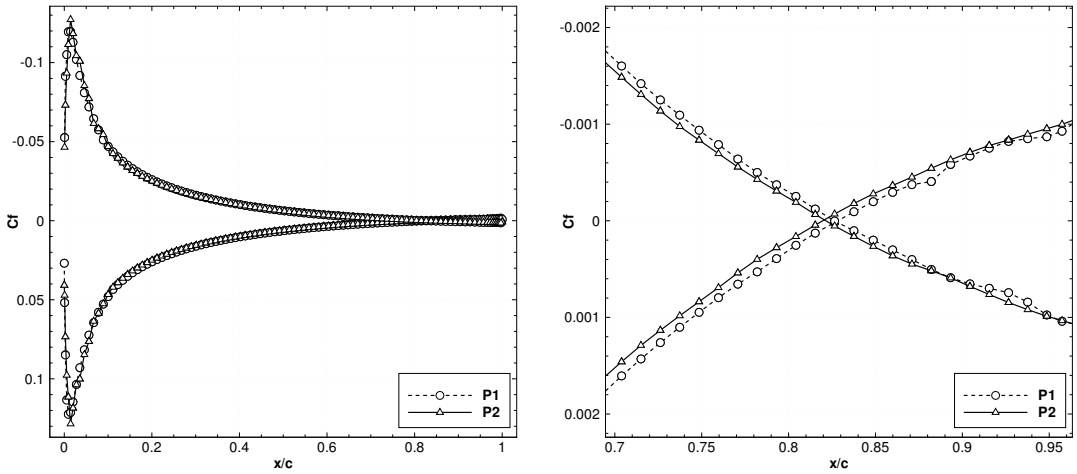


Figure 15: Skin friction coefficient along the whole NACA-0012 airfoil for the second and third order simulations, with the same number of DOFs.

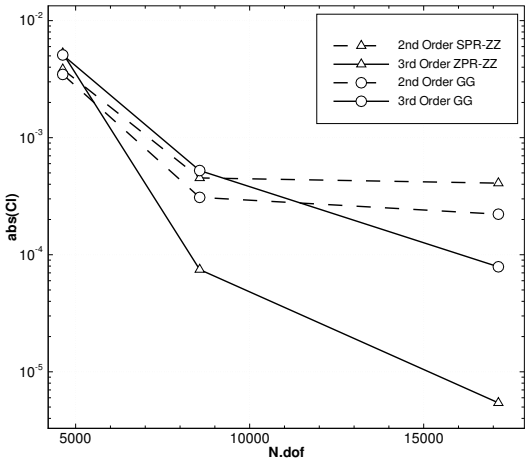


Figure 16: Absolute value of the computed lift coefficients on different grids, with linear and quadratic elements, and with the SPR-ZZ and the Green-Gauss gradient reconstruction methods.

In particular, it is worth noting that, for the same number of DOFs, the simulation with quadratic elements and the SPR-ZZ approach on the finest grid gives an error almost one order of magnitude smaller than that obtained with linear elements and the Green-Gauss gradient reconstruction. Even more important, for a sufficiently small level of error, the third order simulations are about three time less expensive in terms of CPU time than the second order simulations.

6.3. High Reynolds laminar flow over a flat plate

In this test case, a subsonic laminar flow over an adiabatic flat plate is considered. The free-stream Mach number is M = 0.3, the Reynolds number based on the free-stream condition and on the flat plate length is Re = 10^6. The assumption of constant fluid viscosity is used and the Prandtl number is set to Pr = 1. The linear scheme with the SPR-ZZ gradient reconstruction method is used to perform simulations at the second and third order of accuracy.

The range of the computational domain in the x-direction is $[-1, 1]$ and in the y-direction is $[0, 1]$. At the inlet, the free stream condition is imposed. At the top and exit boundaries, the static pressure is fixed. The mesh is clustered in the x and y directions around the leading edge of the flat plate to reduce the effects of the singularity of the solution. Three grids have been used to perform the simulations. The coarsest grid has 19 elements along the plate and 16

	N_{dof}	CD_p	CD_v
SPR-ZZ P1	4 630	0.0220	0.0345
	8 564	0.0222	0.0327
	17 146	0.0224	0.0325
SPR-ZZ P2	4 630	0.0247	0.0369
	8 564	0.0228	0.0324
	17 146	0.0228	0.0324
GG P1	4 630	0.0225	0.0313
	8 564	0.0226	0.0311
	17 146	0.0224	0.0310
GG P2	4 630	0.0251	0.0366
	8 564	0.0230	0.0326
	17 146	0.0229	0.0320

	N_{dof}	CD_p	CD_v
Cell Centered FV [27]	16 384	0.0219	0.0337
Cell Centered FV [27]	65 536	0.0227	0.0327
SD [33] 2nd order	6 912	0.02174	0.03272
SD [33] 3rd order	5 552	0.02219	0.03250
SD [33] 4th order	27 648	0.02225	0.03251
SD [33] 5th order	43 200	0.02225	0.03251
Scheme $k_4 = 1/64$ [22]	40 960	0.0225	0.0344
Scheme $k_4 = 1/128$ [22]	40 960	0.0228	0.0336
Scheme $k_4 = 1/256$ [22]	40 960	0.0229	0.0332

Table 2: Pressure and viscous contributions to the drag coefficient of the NACA-0012 airfoil, for linear and quadratic elements with the SPR-ZZ and the Green-Gauss gradient reconstruction (left) and values reputed in literature (right).

elements in the wall normal direction: the height of the first cell at the wall is about $\Delta y = 2 \times 10^{-4}$. The aspect ratio of the triangles at the wall ranges from a minimum value of 1 : 70 at the leading edge of the plate to a maximum value of 1 : 680 at the end of the plate. Finer grids are obtained with an uniform refinement of the coarsest grid.

In Fig. 18 are reported the distributions of the friction coefficient over the plate obtained with linear and quadratic elements on the different grids. A good agreement with the Blasius solution is obtained even on the coarsest mesh, despite the high stretched elements used. In Fig. 19 and Fig. 20 the comparison with the Blasius solution is made, respectively, for the profiles of the x and y-components of the velocity, at $x = 0.7$. The x-component of the velocity is perfectly computed in all the simulations. In contrast, the vertical component of the velocity is much more difficult to predict and in the outer part of the boundary layer a small difference between the profile of the numerical solution and the Blasius solution can be observed even on the most resolved simulation. Finally, in Fig. 21 the temperature profiles at $x = 0.7$ obtained with the numerical simulations are compared against the theoretical solution of the thermal boundary layer over an adiabatic flat plate for a unitary Prandtl number [31]. An excellent agreement is obtained especially for the third order simulations.

6.4. Laminar flow around a delta wing

In this test case a steady, laminar flow at high angle of attack, around a delta wing with sharp edges is considered. As the flow passes the leading edge, it rolls up and creates a large vortex structure which is convected far behind the wing, at the same time, near the leading edge a smaller secondary vortex appears. A free stream Mach number M = 0.5 is considered, the Reynolds number, based on the root chord of the wing is Re = 4 000, the angle of attack is $\alpha = 12.5°$.

The geometry of the delta wing is depicted in Fig. 22, together with an example of a coarse grid used for the simulations. The grid consists of tetrahedra; finer levels of grids are obtained by uniformly splitting each tetrahedron of the coarser level with eight tetrahedra. The maximum aspect ratio of the elements within the boundary layer over the wing is about 1 : 40. The wing surface is treated as no-slip adiabatic wall, the vertical plane intersecting the root of the wing is treated as a symmetry plane, while far field boundary conditions are applied on the outer boundary of the domain.

The solution is initialized with an uniform flow, the lower order solution is used as initial solution for the third order computation. For this test case the linear scheme is used with the SPR-ZZ gradient reconstruction method; in Fig. 23 are reported the streamlines and Mach number contours, at different stations, of the third order solution on the finest grid. The convergence history is also reported for the second and third order simulations. Note the quadratic convergence in the final stage of the non-linear solver and also the fact that the thanks to the order sequencing strategy, very few iterations are required by the third order method to converge to the steady state.

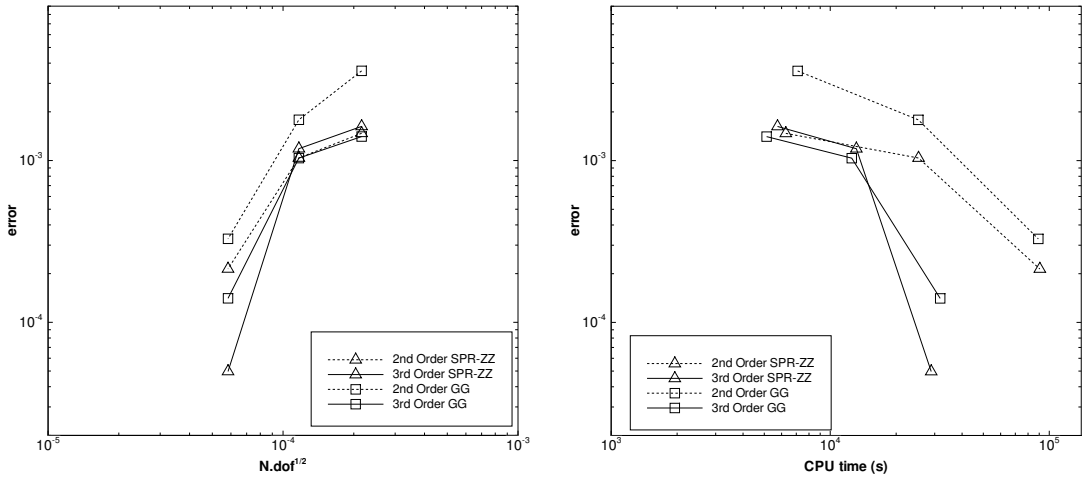


Figure 17: Drag coefficient errors as function of the number of DOF (left) and CPU time in second (right), with linear and quadratic elements, and with the SPR-ZZ and the Green-Gauss gradient reconstruction methods.

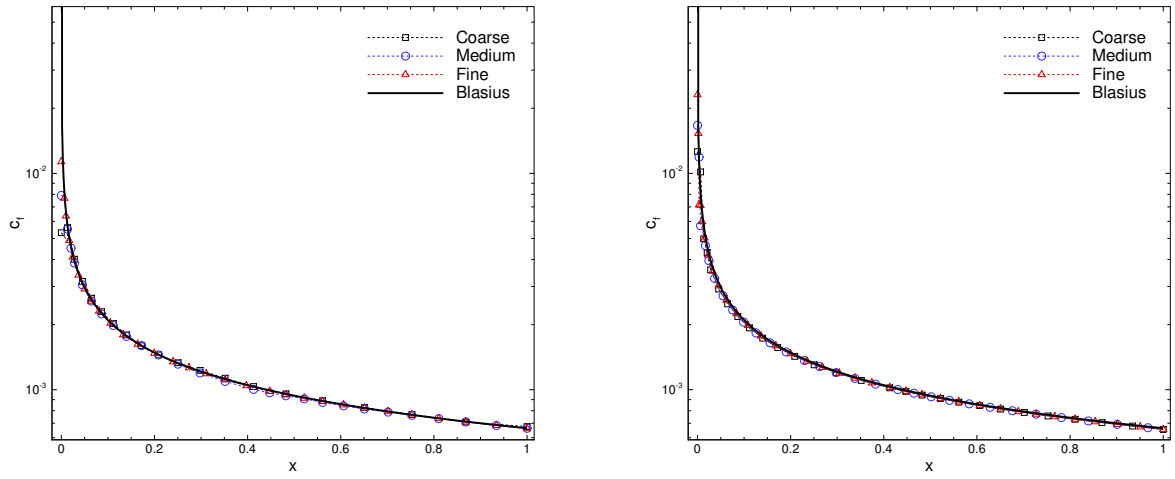


Figure 18: Distribution of the friction coefficient for the high Reynolds number boundary layer over a flat plate computed with linear (left) and quadratic (right) elements.

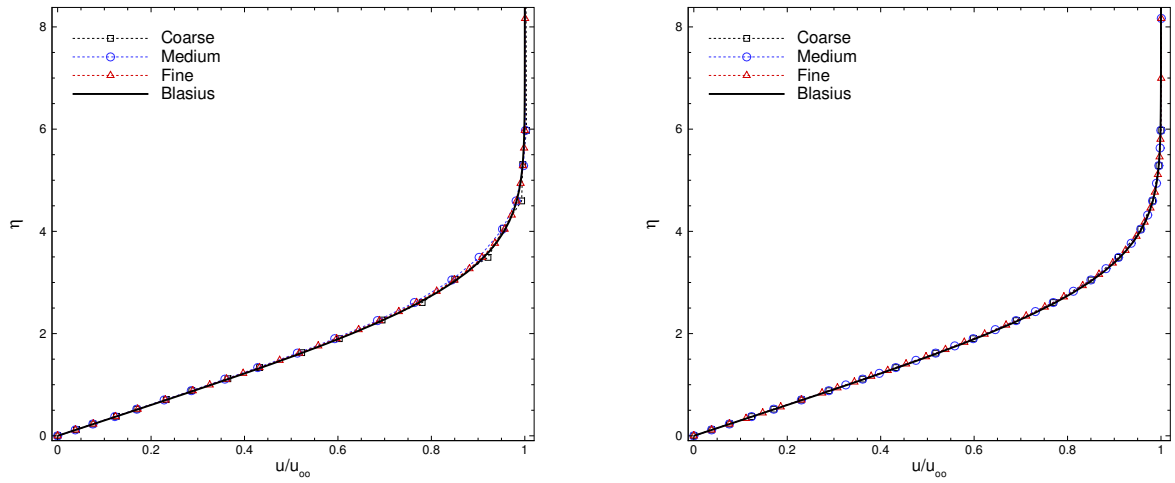


Figure 19: Profile of the x-component of the velocity for the high Reynolds number boundary layer over a flat plate at $x = 0.7$, computed with linear (left) and quadratic (right) elements.

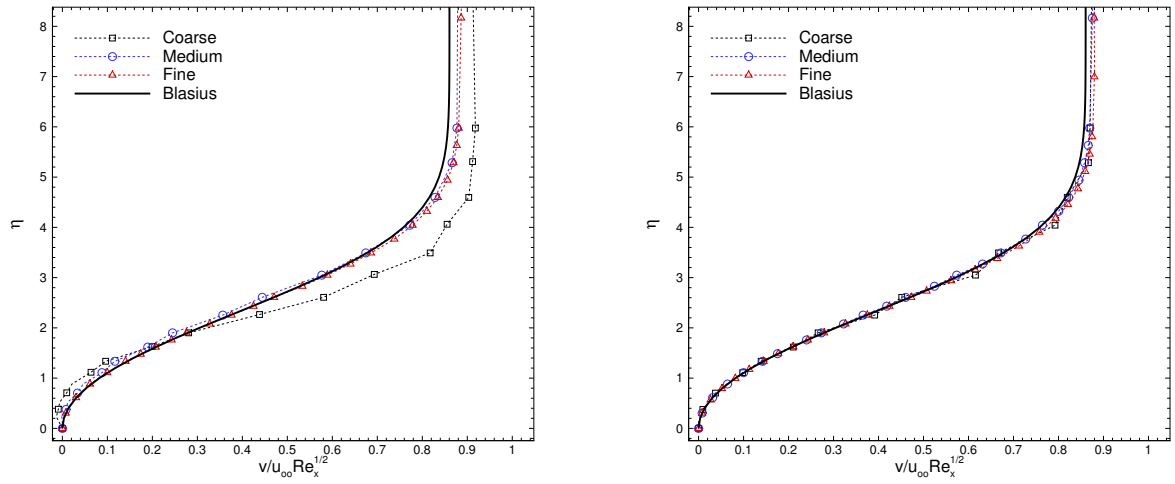


Figure 20: Profile of the y-component of the velocity for the high Reynolds number boundary layer over a flat plate at $x = 0.7$, computed with linear (left) and quadratic (right) elements.

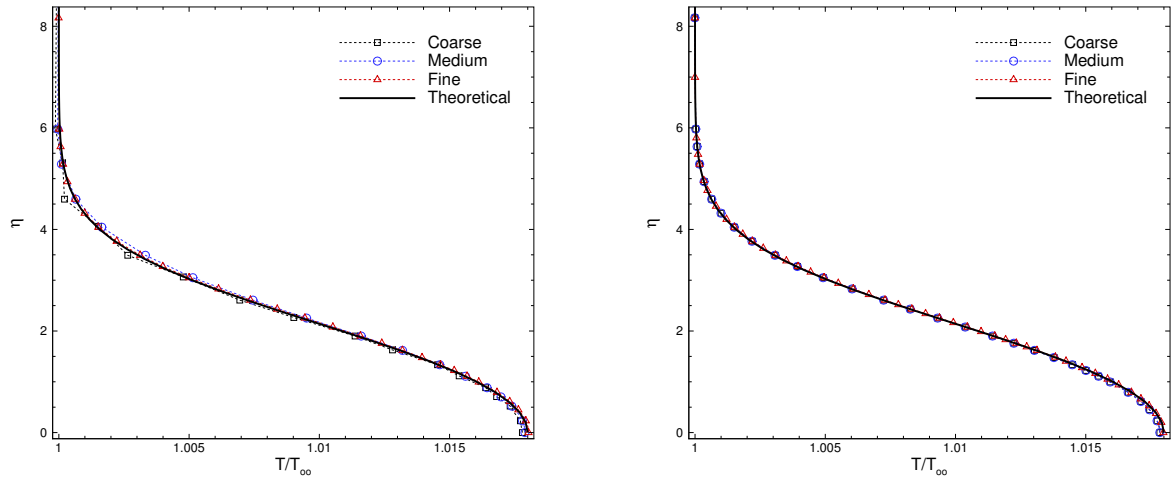


Figure 21: Profile of the temperature for the high Reynolds number boundary layer over a flat plate at $x = 0.7$, computed with linear (left) and quadratic (right) elements.

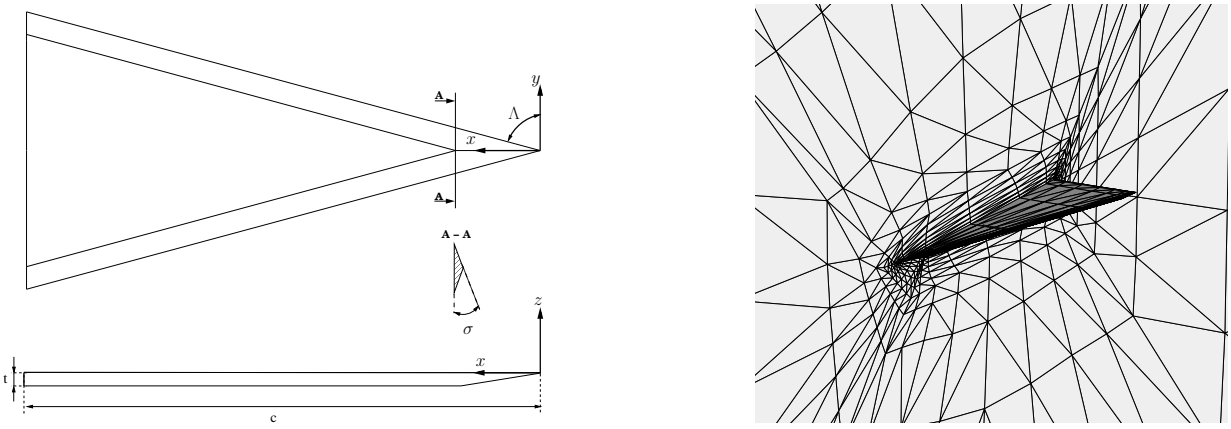


Figure 22: Left: Bottom and side views of the model of the delta wing: $\Lambda = 75°$, $\sigma = 60°$ and $t/c = 0.024$. Right: a coarse mesh of tetrahedra use for the simulations.

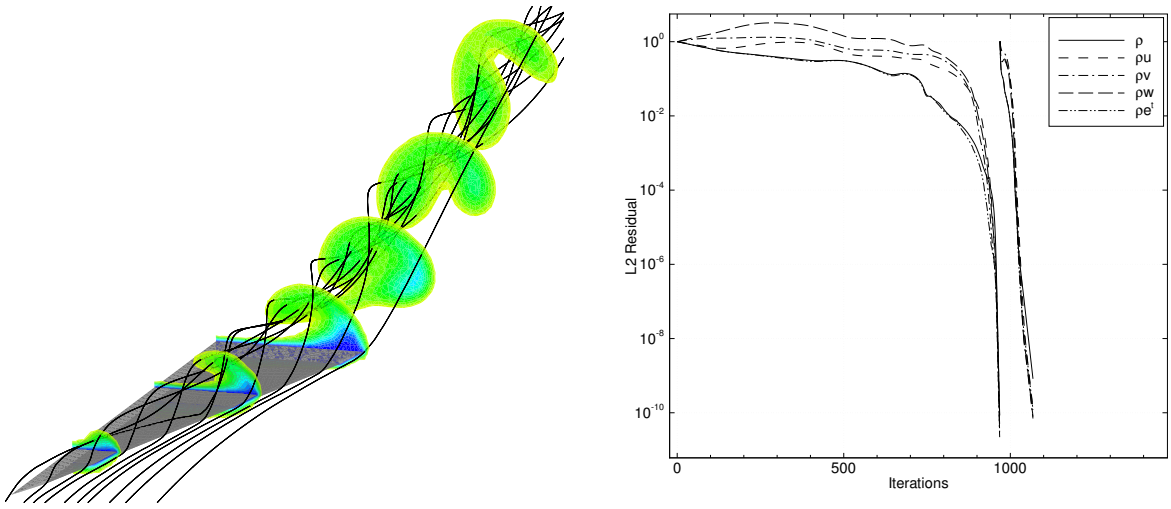


Figure 23: Left: Streamlines and slices of Mach number contours along and behind the delta wing, for a third order simulation on a fine grid. Right: Convergence history for the order sequencing (second and third order).

In Fig. 24 are reported the drag and lift coefficients computed with linear and quadratic elements, on three uniformly refined grids. For comparison, are reported also the reference values computed in [20] by extrapolating the results obtained with a higher order DG method. Observing the convergence of the drag coefficient in term of DOFs, it can be noted that there is no significant gain in using a higher order approximation, with respect to the second order. This behavior can be caused by the singularity at the leading edge of the wing, which might mask the benefits of a higher order approximation with an uniform mesh refinement. Regarding the convergence of the lift coefficient, it could be observed a clear benefit of using a higher order approximation, because the big vortex structure over the wing is better captured with higher order elements. For completeness, in Fig. 25 are reported the errors of the lift and drag coefficients as function of the number of DOFs; the error is computed as the absolute value of the difference between the numerical result and reference value.

6.5. Shock-wave/laminar boundary layer interaction

As last test case, the interaction of an oblique shock wave with a laminar boundary layer is considered. The aim of this test is to verify the non-oscillatory properties of the non-linear scheme in presence of discontinuities of the solution and at the same time, the capability to maintain the accuracy required for the discretization of the boundary layer.

The test consists in a laminar boundary layer developing over a flat plate and an incident shock impinging the boundary layer. Since the flow is supersonic, a shock appears at the leading edge of the flat plate, that interacts with the oblique shock. Furthermore, at the impinging point, the incident shock produces a separation of the boundary layer, the shock is then reflected and an expansion fan appears, turning the flow toward the wall and causing a reattachment of the boundary layer, as it is depicted in Fig. 26.

In the numerical simulations, the oblique shock is generated by imposing the incoming supersonic flow state on the lower part of left boundary, while another supersonic state is imposed on the upper part of the left boundary and on the top boundary; this state is computed using the relations of the oblique shocks, such that the incident shock has a certain angle of incidence θ_s. The height of the computational domain is 0.94, while the range of the computational domain in the x direction is $[-0.2, 2]$, the flat plate has length $L = 2$ with the leading edge of flat plate at $x = 0$. Along the plate, the no-slip adiabatic wall boundary condition is applied, while the symmetry boundary condition is applied on the remaining part of the bottom boundary. On the right boundary, the outflow boundary condition is applied, see Fig. 26. The inflow states are chosen such that the free-stream Mach number is $M = 2.15$ and the angle of the incident shock is $\theta_s = 30.8°$, in this case the impingement point would be at center of the plate for an inviscid fluid. The Reynolds number based on the free-stream values and the distance between the plate leading edge and the inviscid shock impingement point is 1×10^5.

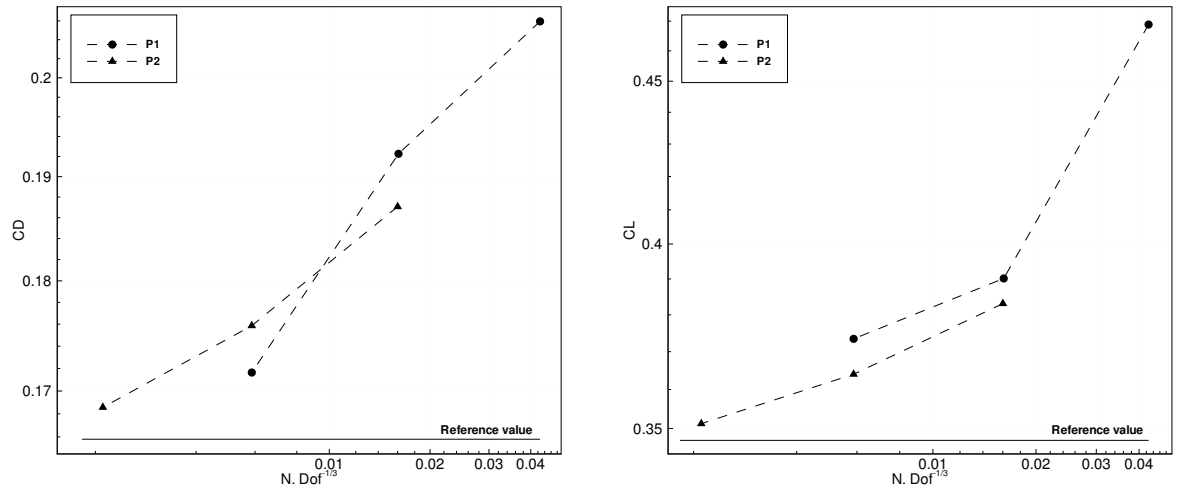


Figure 24: Drag (left) and lift (right) coefficients as function of DOFs for the delta wing simulation, with linear and quadratic elements.

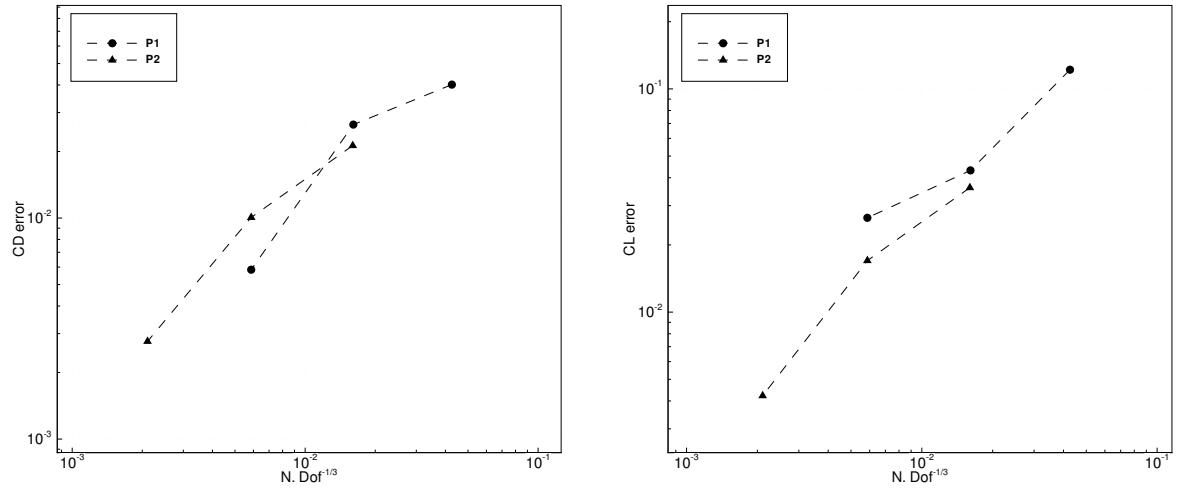


Figure 25: Errors, with respect to the reference values, of the drag (left) and lift (right) coefficients as function of DOFs for the delta wing simulation, with linear and quadratic elements.

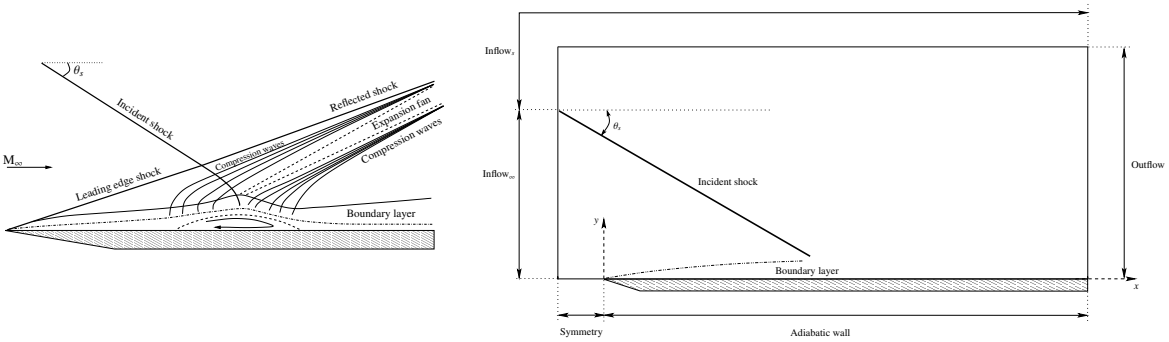
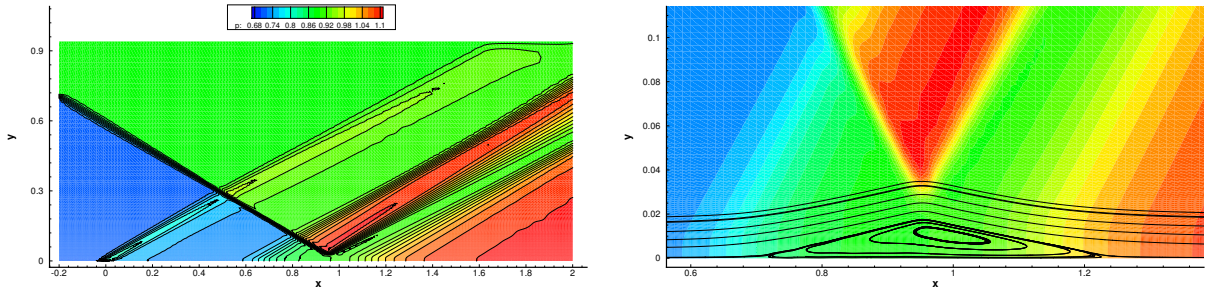


Figure 26: Schematic representation of the waves pattern (left) and computational domain with boundary conditions (right) for the shock-wave/boundary layer interaction problem.



(a) (b)

Figure 27: Left: contours of the pressure obtained with the third order scheme for the shock/boundary layer interaction. Right: zoom of the solution near the impinging point of the shock with the boundary layer, streamlines are also reported to show the separation bubble.

The non-linear scheme with the SPR-ZZ gradient recovery strategy is used to perform the numerical simulations at second and third order of accuracy. The computational domain is generated from the triangulation of a 90×85 structured grid; the first number refers to the number of elements on the horizontal boundaries, with 80 elements along the plates, the second number refers to the number of elements on the vertical boundaries. The element distribution is uniform on the x direction, while along the y direction a non-uniform distribution of the elements is used, with a mesh spacing $\Delta y = 0.5 \times 10^{-3}$ near the bottom boundary. For comparison, a second order simulation is also performed on a finer grid with the same number of DOFs of the third order simulation on the coarse grid. The simulation is initialized with an uniform solution, and the second order solution is used as initial solution for the third order approximation. Except the case of the second order simulation on the coarse grid, for which the initial residual is reduced by ten orders of magnitude, the residual for the third order simulation and the second order one on the finer grid could not be reduced by more than eight orders of magnitude.

In Fig. 27-(a) are shown the contours of the pressure for the third order simulation; all the features of this problem are well represented. In Fig. 27-(b) is reported a zoom of the solution where the incident shock impinges the boundary layer. Two features are evident: the reflection of the incident shock and the recirculation bubble as a consequence of the separation and subsequent reattachment of the boundary layer produced by the incident shock and the expansion fan, respectively.

The profiles of density, pressure and Mach number along the lines at $y = 0.29$ and $y = 0.15$ are reported in Fig. 28. Note that the third order scheme gives a very sharp and monotone representation of the discontinuities and also smooth portions of the solution are better represented compared to the second order solution. It is important to remember that smooth and discontinuous solutions are treated within the same non-linear scheme without any special treatment or tuning parameter. For a fair comparison, it is also reported the solution obtained with the second order scheme on a finer mesh; it is worth noticing that, although a mesh refinement produces an improvement of the numerical solution, the level of accuracy obtained with the second order scheme is still lower than that obtained with the third order scheme, for the same number of DOFs.

Finally, in Fig. 29 are reported the values of the pressure and of the friction coefficient along the plate. The oscillations near the point $x = 0$ are due to the singularity of the solution at the leading edge of the flat plate, but they are limited only in small region around the leading edge. The third order scheme seems less sensitive to this singularity compared to the second order simulations. The separation bubble can easily detected by the negative values of friction coefficients, note also the pressure plateau in the detached zone.

7. Conclusions

A robust and effective Residual Distribution scheme for the discretization of the steady Navier-Stokes equation has been presented. The proposed scheme handles advective and diffusive terms within the same numerical method, making the scheme less sensitive to tuning parameters. A key aspect of the numerical scheme is the reconstruction of the gradient of the numerical solution, in order to have an unique value of the gradients at each degree of freedom of

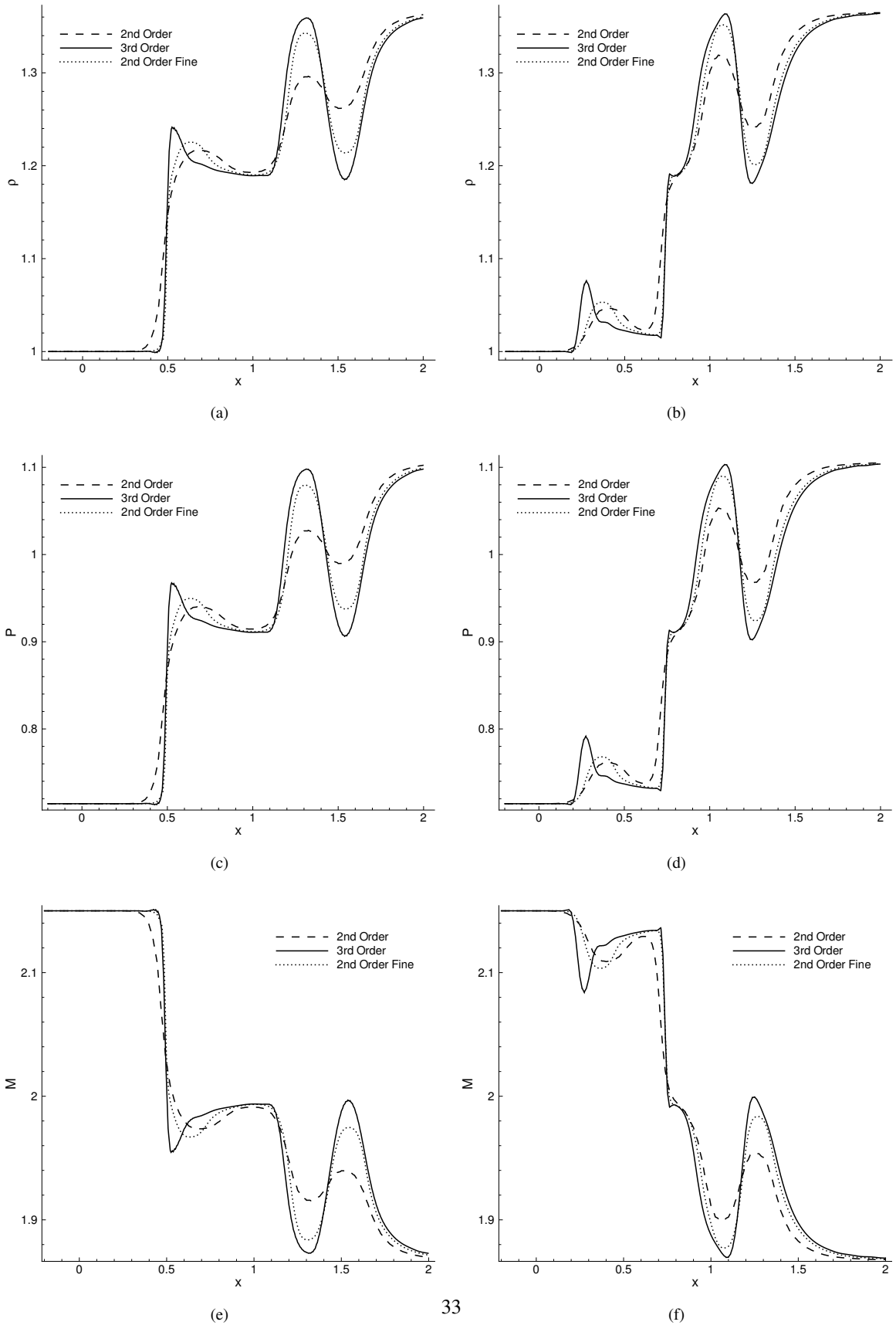
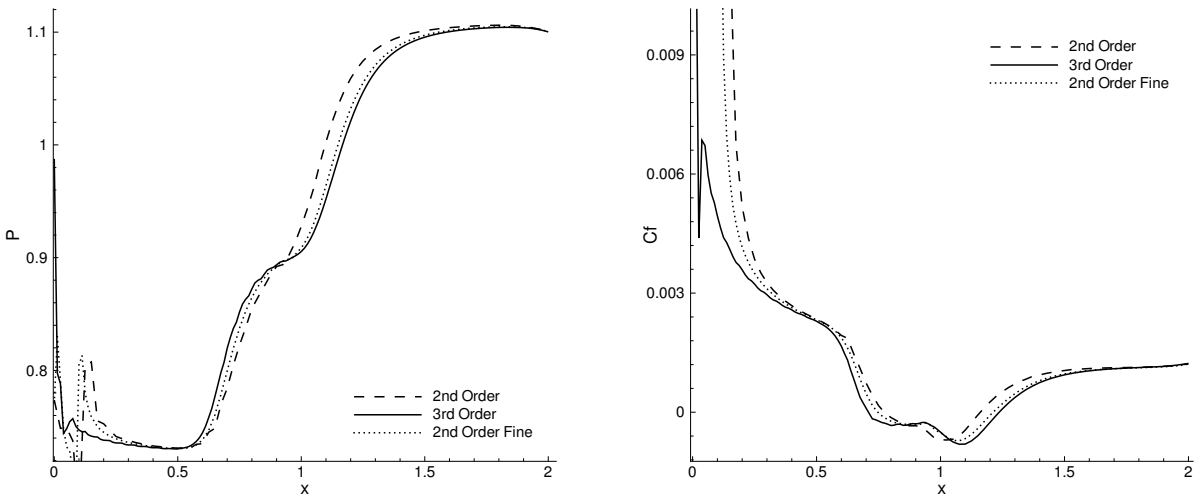


Figure 28: Density, pressure and Mach number profiles along the line $y = 0.29$ (a, c, e) and the line $y = 0.15$ (b, d, f) for the shock/boundary layer interaction problem.



(a) (b)

Figure 29: Pressure (a) and skin friction (b) profiles along the flat plate for the shock/boundary layer interaction problem.

the computational domain. The so-called Super-convergent Patch Recovery method has been used to obtain accurate values of the gradient of the numerical solution, keeping the computational stencil compact. It has been shown that, reconstructing the gradient with high order of accuracy is necessary in order to maintain the theoretical order of approximation of the numerical scheme on the whole domain; for instance the accuracy of the scheme is the same when either the advection or the diffusion dominates, as well as advection and diffusion are equally important.

The use of unstructured grids is essential when the simulation of complex test cases is considered; the proposed method is very flexible, since it is formulated regardless the properties of the grid and the order of approximation of the solution. Furthermore, the possibility to construct non-linear schemes within the Residual Distribution framework, makes these schemes very attractive, because shocks and smooth solution can be handled with the same numerical scheme without introducing any tuning parameter or additional stabilization terms. On the other hand, linear schemes can be still useful when smooth solutions are expected.

The accuracy of the linear and non-linear schemes was tested with the method of the manufactured solution. This test confirmed that a proper reconstruction of the gradient is essential to preserve the accuracy of the numerical scheme. In addition, since the gradient of the numerical solution is important in computation of some quantities (*i.e.*, viscous or thermal stresses on the wall), the capability of computing the numerical solution and its gradient with the same accuracy represents an additional benefit. The flexibility of the numerical method has been tested with more complex test cases, like the two-dimensional flow over a NACA-0012 airfoil and the three-dimensional flow over a delta wing. All the simulations were performed with the implicit Jacobian-free Newton Krylov method, which showed to be very effective and robust in reaching the steady state solution. A shock/boundary layer interaction problem was simulated with the non-linear scheme showing the benefits of using a high-order discretization even for discontinuous flows.

The numerical results presented are very encouraging and as future work the proposed RD approach is going to be formulated for unsteady problems. In the RD approach, time integration is performed by taking into account the residual of the spatial and of the temporal discretization, instead of solving the ODE associated with the semi-discrete spatial discretization. Although, several methods for unsteady problems within the RD framework are available in literature, the majority of these approaches is limited to second order accurate simulations. The objective will be the identification of a general approach that guarantees high order of accuracy for the time discretization.

Acknowledgments.

R.A. has been partially funded by the ERC Advanced Grant "ADDECCO" (contract #226316) and Swiss SNFS grant #200021_153604/1. D.D. has been financed by the FP7 strep "IDIHOM" (contract #265780). We would like to warmly thanks the referees whose comments and criticsms have been very useful

References

[1] R. Abgrall. Essentially non-oscillatory residual distribution schemes for hyperbolic problems. *Journal of Computational Physics*, 214:773–808, 2006.

[2] R. Abgrall. A residual distribution method using discontinuous elements for the computation of possibly non smooth flows. *Advances in Applied Mathematics and Mechanics*, 2:32–44, 2010.

[3] R. Abgrall, D. De Santis, and M. Ricchiuto. High order preserving residual distribution schemes for advection-diffusion scalar problems on arbitrary grids. *SIAM Journal on Scientific Computing*, 36(3):955–983, 2014.

[4] R. Abgrall, A. Larat, and M. Ricchiuto. Construction of very high order residual distribution schemes for steady inviscid flow problems on hybrid unstructured meshes. *Journal of Computational Physics*, 230:4103–4136, 2011.

[5] R. Abgrall, A. Larat, M. Ricchiuto, and C. Tavé. A simple construction of very-high order non oscillatory compact schemes on unstructured meshes. *Computers and Fluids*, 38(7):1314–1323, 2009.

[6] R. Abgrall and P. L. Roe. High-order fluctuation schemes on triangular meshes. *Journal of Scientific Computing*, 19:3–36, 2003.

[7] R. Abgrall and D. De Santis. High order residual distribution scheme for Navier-Stokes equations. In *20th AIAA Computational Fluid Dynamics Conference*. AIAA 2011-3231, 2011.

[8] R. Abgrall and J. Trefilik. An example of high order residual distribution scheme using non-lagrange elements. *Journal of Scientific Computing*, 45(1-3):3–25, 2010.

[9] P.N. Brown and Y. Saad. Hybrid Krylov methods for nonlinear systems of equations. *SIAM Journal on Scientific and Statistical Computing*, 11:450–481, 1990.

[10] D. Caraeni and L. Fuchs. Compact third-order multidimensional upwind scheme for Navier–Stokes simulations. *Theoretical and Computational Fluid Dynamics*, 15:373–401, 2002.

[11] B. Cockburn, G.E. Karniadakis, and C.W. Shu. *Discontinuous Galerkin methods: theory, computation and application*. Lecture notes in computational science and engineering. Springer, Berlin, 2000.

[12] Bernardo Cockburn and Chi-Wang Shu. The Local Discontinuous Galerkin Method for Time-Dependent Convection-Diffusion Systems. *SIAM Journal on Numerical Analysis*, 35(6):2440, 1998.

[13] Herman Deconinck and Mario Ricchiuto. Residual Distribution Schemes: Foundations and Analysis. In *Encyclopedia of Computational Mechanics*. John Wiley & Sons, Ltd, 2004.

[14] Michael Dumbser. Arbitrary high order PN-PM schemes on unstructured meshes for the compressible NavierStokes equations. *Computers & Fluids*, 39(1):60–76, 2010.

[15] Gregor Gassner, Frieder Lörcher, and Claus-Dieter Munz. A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes. *Journal of Computational Physics*, 224(2):1049–1063, 2007.

[16] L. R. Herrmann. Interpretation of finite element procedure as stress error minimization procedure. *Journal of the Engineering Mechanics Division*, 98(5):1330–1336, 1972.

[17] Thomas J. R. Hughes. Recent progress in the development and understanding of SUPG methods with special reference to the compressible Euler and Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 7(11):1261–1275, 1987.

[18] A. Jameson and S. Yoon. Lower-upper implicit schemes with multiple grids for the Euler equation. *AIAA Journal*, 25:929–935, 1987.

[19] D. A. Knoll and D. Keyes. Jacobian-free Newton method: a survey of approaches and application. *Journal of Computational Physics*, 193:357–397, 2004.

[20] T. Leicht and R. Hartmann. Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations . *Journal of Computational Physics*, 229(19):7344 – 7360, 2010.

[21] Hong Luo, Luqing Luo, Robert Nourgaliev, Vincent A. Mousseau, and Nam Dinh. A reconstructed discontinuous Galerkin method for the compressible NavierStokes equations on arbitrary grids. *Journal of Computational Physics*, 229(19):6961–6978, 2010.

[22] D. J. Mavriplis, A. Jameson, and L. Martinelli. Multigrid solution of the Navier-Stokes equations on triangular meshes. AIAA Paper 89-0120, 1989.

[23] R.-H. Ni. A multiple grid scheme for solving the Euler equations. In *5th Computational Fluid Dynamics Conference*, pages 257–264, 1981.

[24] H. Nishikawa. Robust and accurate viscous discretization via upwind scheme I: Basic principle. *Computers & Fluids*, 49:62–86, 2011.

[25] H. Nishikawa and P. L. Roe. On High-Order Fluctuation-Splitting Schemes for Navier–Stokes Equations. In *Computational Fluid Dynamics 2004: Proceedings of the Third International Conference on Computational Fluid Dynamics, ICCFD, Toronto, 12-16 July 2004*. Springer 2006, 2004.

[26] M. Pernice and H.F. Walker. NITSOL: A Newton iterative solver for nonlinear systems. *SIAM Journal on Scientific and Statistical Computing*, 19:302–318, 1998.

[27] R. Radespiel and R.C. Swanson. An investigation of cell-centered and cell vertex multigrid schemes for Navier-Stokes equations. AIAA Paper 89-0543, 1989.

[28] M. Ricchiuto, N. Villedieu, R. Abgrall, and H. Deconinck. On uniformly high-order accurate residual distribution schemes for advection-diffusion. *Journal of Computational and Applied Mathematics*, 215(2):547 – 556, 2008.

[29] C.J. Roy, C. C. Nelson, and T. M. Smith. Verification of Euler/NavierStokes codes using the method of manufactured solutions. *International Journal for Numerical Methods in Fluids*, 44(6):599–620, 2004.

[30] T. Saad and M.H. Schultz. GMRES: a generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 7:856–869, 1986.
[31] H. Schlichting and K. Gersten. *Boundary-layer theory*. Springer, 2000.
[32] V. Selmin, J. Donea, and L. Quartapelle. Finite element methods for nonlinear advection . *Computer Methods in Applied Mechanics and Engineering*, 52(1–3):817 – 845, 1985.
[33] Y. Sun, Z.J. Wang, and Y. Lun. Efficient Implicit Non-linear LU-SGS Approach for Compressible Flow Computation Using High-Order Spectral Difference Method. *Communications in Computational Physics*, 5(2-4):760–778, 2009.
[34] K. R. Jackson T. F. Chan. Nonlinearly preconditioned Krylov subspace methods for discrete Newton algorithms. *SIAM Journal on Scientific and Statistical Computing*, 5:533–4542, 1984.
[35] M.D. Tidiriri. Preconditioning techniques for the Newton-Krylov solution of compressible flows. *Journal of Computational Physics*, 132:51–61, 1997.
[36] B. van Leer and S. Nomura. Discontinuous Galerkin for diffusion. In *Proceedings of 17th AIAA computational fluid dynamics conference*. AIAA-2005-5108, 2005.
[37] Z.J. Wang, Krzysztof Fidkowski, Rémi Abgrall, Francesco Bassi, Doru Caraeni, Andrew Cary, Herman Deconinck, Ralf Hartmann, Koen Hillewaert, H.T. Huynh, Kroll Norbert, Georg May, Per-Olof Persson, Bram van Leer, and Miguel Visbal. High-order CFD methods: current status and perspective. *International Journal for Numerical Methods in Fluids*, 72(8):811–845, 2013.
[38] O. C. Zienkiewicz and J. Z. Zhu. A simple error estimator and adaptive procedure for practical engineerng analysis. *International Journal for Numerical Methods in Engineering*, 24(2):337–357, 1987.
[39] O. C. Zienkiewicz and J. Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. Part 2: Error estimates and adaptivity. *International Journal for Numerical Methods in Engineering*, 33(7):1365–1382, 1992.
[40] O.C. Zienkiewicz and R.L.Taylor. *Finite Element Method (5th Edition) Volume 1 - The Basis*. Elsevier, 2000.

Appendix A. Least-square gradient reconstruction

To reconstruct the gradient of the numerical solution at the node i of the grid, the solution is expanded in a Taylor series around the node i for each node j belonging to the stencil of i,

$$u_j = u_i + \left.\frac{\partial u}{\partial x}\right|_i (x_j - x_i) + \left.\frac{\partial u}{\partial y}\right|_i (y_j - y_i) + \left.\frac{\partial^2 u}{\partial x^2}\right|_i (x_j - x_i)^2 + \left.\frac{\partial^2 u}{\partial y^2}\right|_i (y_j - y_i)^2 + \left.\frac{\partial^2 u}{\partial x \partial y}\right|_i (x_j - x_i)(y_j - y_i) + \dots,$$

where $u_i = u(\boldsymbol{x}_i)$ and $u_j = u(\boldsymbol{x}_j)$. The gradient reconstruction is obtained by solving for the values of the gradient that minimize the following function

$$\sum_{j=1}^{N_i} \omega_{ij}^2 E_{ij}^2, \qquad \forall\ i \in \mathcal{N}_h$$

with

$$E_{ij}^2 = \left(-\Delta u_{ij} + \left.\frac{\partial u}{\partial x}\right|_i \Delta x_{ij} + \left.\frac{\partial u}{\partial y}\right|_i \Delta y_{ij} + \left.\frac{\partial^2 u}{\partial x^2}\right|_i \Delta x_{ij}^2 + \left.\frac{\partial^2 u}{\partial y^2}\right|_i \Delta y_{ij}^2 + \left.\frac{\partial^2 u}{\partial x \partial x}\right|_i \Delta x_{ik} \Delta y_{ij} + \dots \right)^2,$$

where $\Delta u_{ij} = u_j - u_i$, $\Delta x_{ij} = x_j - x_i$, $\Delta y_{ij} = y_j - y_i$, while ω_{ij} is a weight factor. In the case of linear elements, the solution is expanded only up to the first derivatives in the Taylor series, while in the case of quadratic elements second derivatives are also included in the Taylor expansion and the minimization is done with respect to first and second derivatives. In compact form, the minimization problem can written as follows

$$\boldsymbol{A}_i \boldsymbol{d}_i = \boldsymbol{b}_i, \qquad \boldsymbol{A}_i \in \mathbb{R}^{M_i \times N}, \boldsymbol{d}_i \in \mathbb{R}^N, \boldsymbol{b}_i \in \mathbb{R}^{M_i}, \tag{A.1}$$

where N is the number of partial derivatives of u, and $M_i (\geq \mathrm{N})$ is the number of neighboring nodes of the node i. In the case of quadratic elements, for example, the terms in the previous linear system read

$$\boldsymbol{A}_i = \begin{pmatrix} (x_1 - x_i) & (y_1 - y_i) & (x_1 - x_i)(y_1 - y_i) & (x_1 - x_i)^2 & (y_1 - y_i)^2 \\ (x_2 - x_i) & (y_2 - y_i) & (x_2 - x_i)(y_2 - y_i) & (x_2 - x_i)^2 & (y_2 - y_i)^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (x_{M_i} - x_i) & (y_{M_i} - y_i) & (x_{M_i} - x_i)(y_{M_i} - y_i) & (x_{M_i} - x_i)^2 & (y_{M_i} - y_i)^2 \end{pmatrix}$$

$$\boldsymbol{d}_i = \begin{pmatrix} \left.\frac{\partial u}{\partial x}\right|_i \\ \left.\frac{\partial u}{\partial y}\right|_i \\ \left.\frac{1}{2}\frac{\partial^2 u}{\partial xy}\right|_i \\ \left.\frac{1}{2}\frac{\partial^2 u}{\partial x^2}\right|_i \\ \left.\frac{1}{2}\frac{\partial^2 u}{\partial y^2}\right|_i \end{pmatrix}, \quad \text{and} \quad \boldsymbol{b}_i = \begin{pmatrix} u_1 - u_i \\ u_2 - u_i \\ \vdots \\ u_{M_i} - u_i \end{pmatrix}.$$

The solution of the Eq. (A.1) must be sought in the least-squares sense. The weight factor ω_{ij} is generally taken as the inverse of the distance between the nodes i and j.